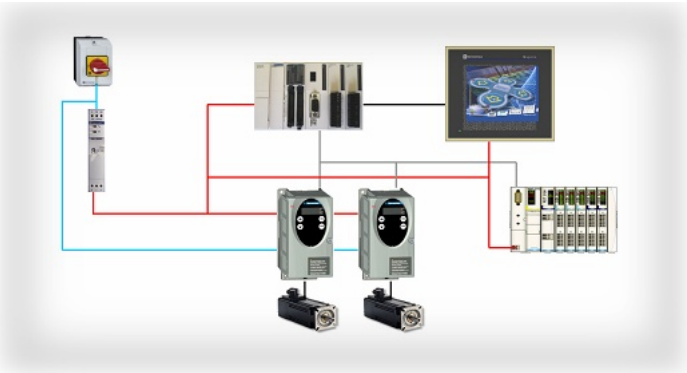
Twido and Lexium Magelis and Advantys

System User Guide

[source code]



33003565.04

Merlin Gerin
Square D
Telemecanique

Schneider Electric
Building a New Electric World

SEP 2006

Contents

Application Source Code 4
Typical applications 5
System 6
Architecture 6
Installation 9
Hardware 10
Software 18
Communication 19
Implementation 26
Communication 27
HMI 32
PLC 49
Devices 66
Performance 86
Appendix 87
Detailed Component List 87
Component Protection Classes 88
Component Features 89
Contact 91

Introduction

This document is intended to provide a quick introduction to the described System.
It is **not** intended to replace any specific product documentation. On the contrary, it offers additional information to the product documentation, for installing, configuring and starting up the system.

A detailed functional description or the specification for a specific user application is **not** part of this document. Nevertheless, the document outlines some typical applications where the system might be implemented.

Abbreviations

Word / Expression	Signification
AC	Alternating Current
Advantys	SE product name for a family of I/O modules
Altivar (ATV)	SE product name for a family of VSDs
CANopen	Name for a communications machine bus system
CB	Circuit Breaker
CoDeSys	Hardware-independent IEC 61131-3 programming software
ConneXium	SE product name for a Family of Transparent Factory devices
DC	Direct Current
EDS	Electronic Data Sheet
E-OFF, E-STOP	Emergency Off switch
Harmony	SE product name for a family of switches and indicators
HMI	Human Machine Interface
I/O	Input/Output
IclA (ICLA)	SE product name for a compact drive
Lexium/Lexium05/LXM	SE product name for a family of servo-drives
Magelis	SE product name for a family of HMI-Devices
MB - SL	SE name for a serial Modbus communications protocol
Micro	SE product name for a middle range family of PLCs
NIM	SE product name for a Network Interface Module
Osiswitch	SE product name for a family of position switches
PC	Personal Computer
Phaseo	SE product name for a family of power supplies
PLC	Programmable Logic Computer
Powersuite	An SE software product for configuring drives
Premium	SE product name for a middle range family of PLCs
Preventa	SE product name for a family of safety devices
PS1131 (CoDeSys)	SE Product name for PLC programming software with CoDeSys
PS	Power Supply
SE	Schneider Electric
Sycon	SE product name of a Field bus programming software
Telefast	SE product name for a series of distributed I/O devices
TesysU	SE product name for a decentralised I/O System
Twido	SE product name of a middle range family of PLCs
TwidoSoft	SE product name for a PLC programming software
TwidoSuite	SE product name for a PLC programming software
Unity (Pro)	SE product name for a PLC programming software
Vijeo Designer	An SE software product for programming Magelis HMI devices
VSD	Variable Speed Drive
WxHxD	Dimensions : Width, Height and Depth
XBT-L1000	An SE software product for programming Magelis HMI devices
Zelio	SE product name for a low range PLC family
ZelioSoft	SE product name for a PLC programming software

Application Source Code

Introduction

Examples of the source code used to attain the system function as described in this document can be downloaded from our website under this link.

The example source code is in the form of configuration, application and import files. Use the appropriate software tool to either open or import the files.

Extension	File Type	Software Tool Required
AIW	Configuration file	Advantys
CNF	Configuration File	Sycon
CO	CANopen definitions file	Sycon
CSV	Comma Seperated Values, Spreadsheet	Twidosoft
CTX		Unity
DCF	Device Configuration File	Advantys
DIB	Device Independent Bitmap	Sycon
DOC	Document file	Microsoft Word
DOP	Project File	Magelis XBTL
EDS	Electronic Data Sheet – Device Definition	Industrial standard
FEF	Export file	PL7
GSD	EDS file (Geraete Stamm Datei)	Profibus
ISL	Island file, project file	Advantys
PB	Profibus definitions file	Sycon
PDF	Portable Document Format - document	Adobe Acrobat
PRO	Project file	PS1131 - CoDeSys
PS2	Export file	Powersuite
RTF	Rich Text File - document	Microsoft Word
SPA	Schneider Product Archive	TwidoSuite
STU	Project file	Unity Pro
STX	Project file	PL7
TLX	Project file	Twinline control tool
TWD	Project file	TwidoSoft
VDZ	Project file	Vijeo Designer
XEF	Export file	Unity Pro
XPR	Project file	TwidoSuite
ZM2	Project file	Zeliosoft

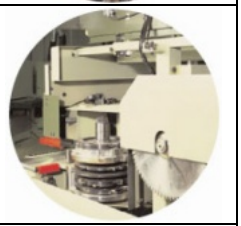
Typical applications

Introduction

Here you will find a list of the typical applications, and their market segments, where this system or subsystem can be applied:

Industrial
- Small to middle sized automated machines
- De-centralised automated sub systems serving as components in large or middle sized machines

Machines
- Packaging Machines and box folders
- Conveyors

Application	Description	Example
Packaging machines	In the packaging industry, for labeling, packaging, filling and palletizing goods	
Special-purpose machines	Used on small special-purpose machines for assembly, processing, cutting operations, etc. (e.g. food preparation, automated assembly, wood machining).	
Material conveyors	Used in connection with transportation tasks that involve “pick and place” operations.	

System

Introduction

This section describes the architecture, components, size and number of the components that are used within this system.

Architecture

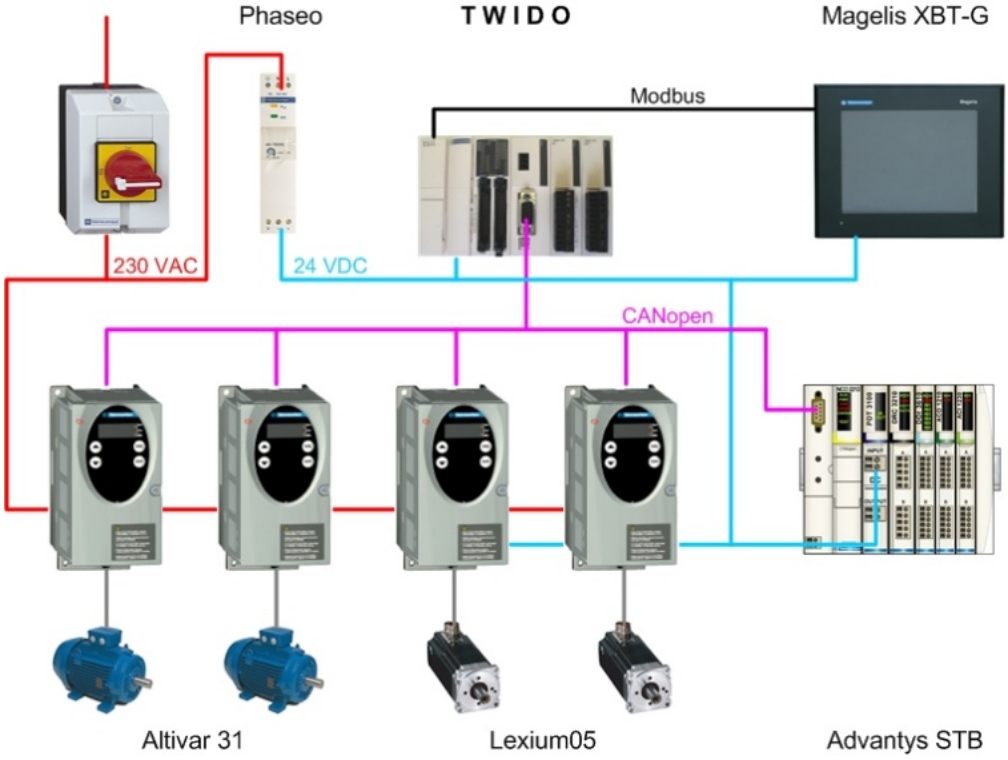
Overview

The control section of this application consists of a Twido PLC servicable via a Magelis HMI user display.

The PLC supervises Lexium05 servo drives and Altivar 31 controllers. The system contains an Advantys STB I/O platform.

System communcations are implemented with CANopen and Modbus.

Layout



Components

Hardware:

- Twido (PLC)
- Phaseo (power supply)
- Lexium05 (drive control)
- Altivar 31 (motor drive)
- Advantys STB (remote I/O)
- Magelis XBTG (HMI)

Software:

- TwidoSoft 3.2 (PLC)
- Advantys configuration software ver. 2.00 (remote I/O)
- Vijeo-Designer ver. 4.3 (HMI)
- PowerSuite 2.3 (Lexium05 and Altivar 31)

Quantities of Components

For a complete and detailed list of components, the quantities required and the order numbers, please refer to the components list at the rear of this document.

Degree of Protection

Not all the components in this configuration are designed to withstand the same environmental conditions. Some components may need additional protection, in the form of housings, depending on the environment in which you intend to use them. For environmental details of the individual components please refer to the list in the appendix of this document and the appropriate user manual.

Technical Data

Mains voltage	230V AC
Power requirement	~ 3.0 kW
Drive power rating	2x 0,75 kW, 2x 0,18 kW
Motor brake	none
Connection	3 x 2,5mm² (L1, L2, L3, N, PE)
Safety Level	none

Safety Notice

The standard and level of safety you apply to your application is determined by your system design and the overall extent to which your system may be a hazard to people and machinery.

As the aim of this document is to detail the bus communication between the individual devices, no safety category has been selected or recommended.

Whether or not the above safety category should be applied to your system should be ascertained with a proper risk analysis.

This document is not comprehensive for any systems using the given architecture and does not absolve users of their duty to uphold the safety requirements with respect to the equipment used in their systems or of compliance with either national or international safety laws and regulations

Dimensions

The dimensions of the individual devices used; PLC, Drive, Power supply, etc. require a housing cabinet size of at least 1000x600x300 mm (WxHxD).

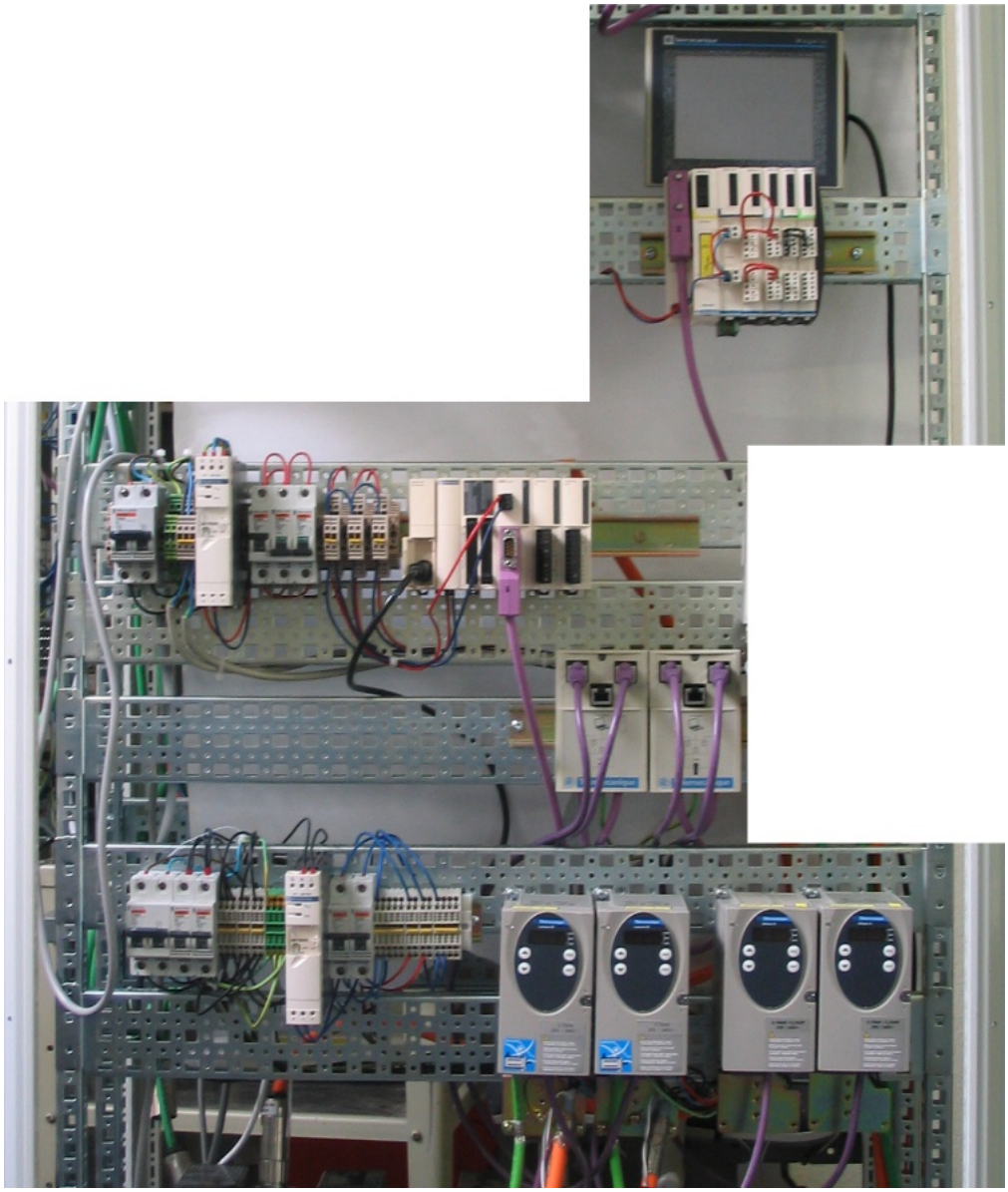
The HMI display and illuminated indicators can be built into the door of the housing.

Installation

Introduction

This chapter describes the steps necessary to set up the hardware and configure the software required to fulfil the described function of the application.

Layout



Note

The configuration of this application has not been developed for any special actual use. It is intended to show how the system components work together as a unit. The components that are listed are a cross-section of the components needed for control and display in possible applications.

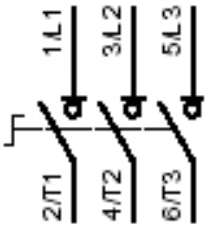
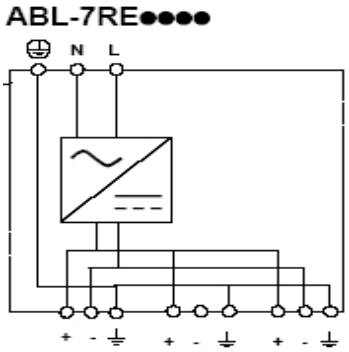
This document does not claim to be comprehensive and **does not absolve users** from their duty to check the safety requirements of their equipment and to ensure compliance with the relevant national or international standards and regulations.

Hardware

General

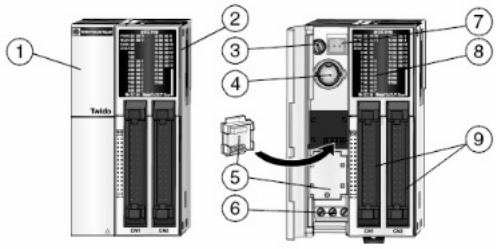
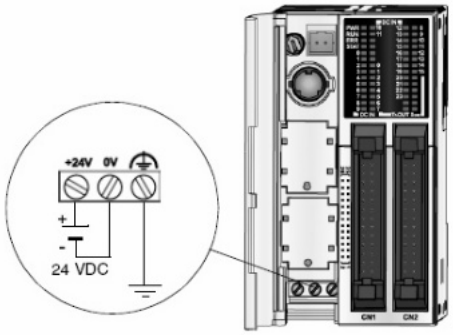
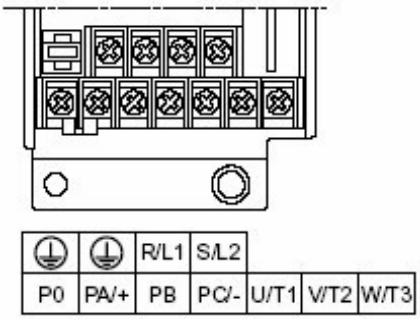
- For assembly purposes, the Twido, power supply and Advantys STB require a top-hat rail.
- The other devices can be attached directly to the mounting plate.
- A 230 V AC wiring is used between the main switch, power supply and VSD.
- A 24 V DC wiring is used between the power supply, PLC, HMI and VSD control unit.
- There are other cables from the power cables (ATV and LXM) and feedback cables (only LXM) between the motor and the VSD.

Components

Master switch **VCF-02GE**		
Power supply **ABL7RE2403**		

Continued on next page

Components
contd.

Twido PLC **TWDLMDA40DTK**		 1 Hinged cover 2 Extension connector 3 Analog potentiometer 4 Serial port 1 5 Module cover 6 24 V DC power supply terminals 7 Analog voltage supply connector 8 LEDs 9 I/O terminals
Twido PLC **TWDLMDA40DTK** **Power supply**		
Drive Controller **ATV31** incl. PS and motor connection **ATV31H018M2**		

Drive Controller

ATV31

incl. PS and motor connection

ATV31H018M2

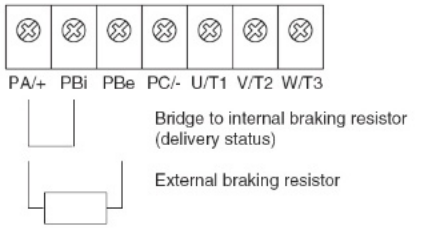
Connections

Terminal	Function
⏚	Ground terminal
R/L1 S/L2	Power supply
R/L1 S/L2 T/L3	
PO	DC bus + polarity
PA/+	Output to braking resistor (+ polarity)
PB	Output to braking resistor
PC/-	DC bus - polarity
U/T1 V/T2 W/T3	Outputs to the motor

Lexium05

Servo Drive

LXM05AD10M2



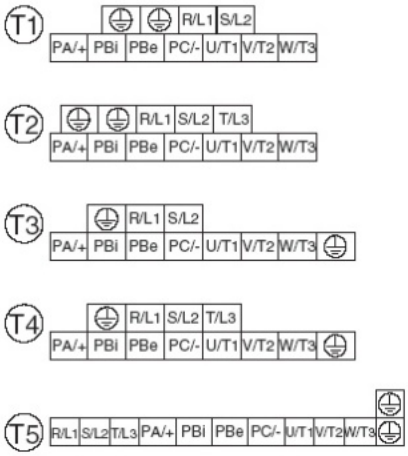
Power connections	Description
PE	Earth connection (protective earth)
R/L1, S/L2/N	Mains connection, single phase devices
R/L1, S/L2, T/L3	Mains connection, 3-phase devices
PA/+	DC bus
PBi	Braking resistor internal
PBe	Braking resistor external
PC/-	DC bus
U/T1,V/T2, W/T3	Motor connections

Lexium05

Servo Drive

LXM05AD10M2

Power connections (see T1)



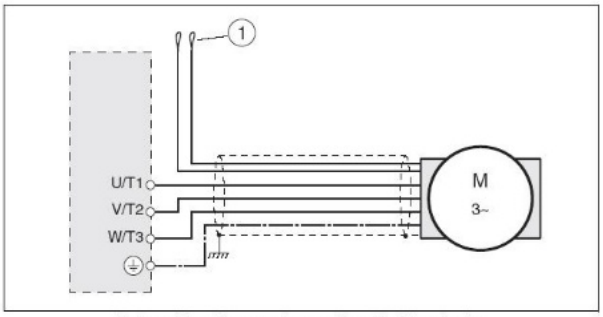
LXM05•...	
D10F1	(T1)
D10M2	(T1)
D10M3X	(T2)
D14N4	(T4)
D17F1	(T3)
D17M2	(T4)
D17M3X	(T4)
D22N4	(T4)
D28F1	(T3)
D28M2	(T4)
D34N4	(T4)
D42M3X	(T4)
D57N4	(T5)

Lexium05

Servo Drive

LXM05AD10M2

Connection for Motors



Motor wiring diagram, here without holding brake

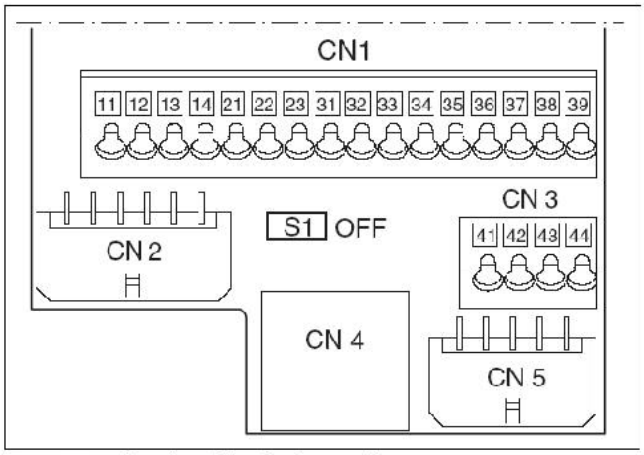
Terminal	Description	Colour
U/T1	Motor lead	black L1 (BK)
V/T2	Motor lead	black L2 (BK)
W/T3	Motor lead	black L3 (BK)
PE	Protective conductor	green/yellow (GN/YE)
(1)	Holding brake connection cable For motors with holding brake	white (WH), grey (GR)

Lexium05

Servo Drive

LXM05AD10M2

Overview of signal connections



Overview of the signal connections

Connection/ switch	Assignments
CN1	Analogue inputs ±10V, pin 11 to 14
	CANopen, pin 21-23
	Digital inputs/outputs, pin 31-39
CN2	Motor encoder (Hiperface Sensor)
CN3	24V PELV controller supply voltage
CN4	PC, peripheral operating terminal, Modbus, CANopen;(RJ45)
CN5	ESIM (A/B/I out), PULSE/DIR in, encoder signals A/B/I in [1)]
S1	Switch for fieldbus terminating resistor

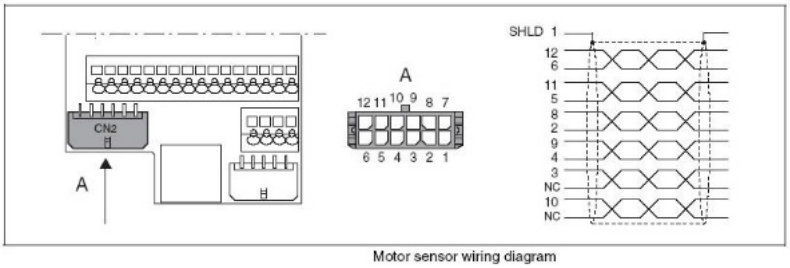
1) depending on the "First Setup"

Lexium05

Servo Drive

LXM05AD10M2

Motor encoder



Motor sensor wiring diagram

Pin	Signal	Motor, pin	Colour [1]	Pair	Description	I/O
1	SHLD				Shielding braid	
12	SIN	8	white	1	Sine signal	E
6	REFSIN	4	brown	1	Reference for sine signal, 2.5 V	A
11	COS	9	green	2	Cosine signal	E
5	REFCOS	5	yellow	2	Reference for cosine signal, 2.5V	A
8	Data	6	grey	3	Receive and transmit data	I/O
2	$\overline{\text{Data}}$	7	pink	3	Receive and transmit data, inverted	I/O
10	ENC_0V	11	blue	4	sensor reference potential (encoder) (0.5mm²)	A
			red	4	not assigned (0.5mm²)	
3	TMOT_0V	1	black	5	Reference potential for T_MOT	-
			purple	5	not assigned	
9	T_MOT	2	grey/pink	6	temperature sensor PTC	E
4	ENC+10V_OUT	10	red/blue	6	10 V_{DC} power supply for sensor, max. 150 mA	A
7	n.c.				not assigned	

1) Colour data is based on the prefabricated cables

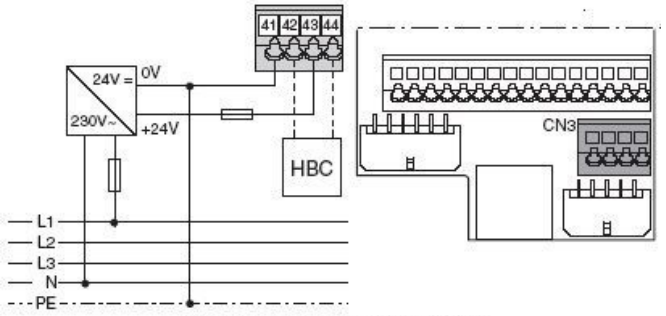
Lexium05

Servo Drive

LXM05AD10M2

24V power supply

HBC
Holding brake control



Controller supply connection diagram

Pin	Signal	Description
41	0VDC	Reference potential for 24V voltage
42	0VDC	Reference potential for 24V voltage
43	+24VDC	24V controller supply voltage
44	+24VDC	24V controller supply voltage

Lexium05

Servo Drive

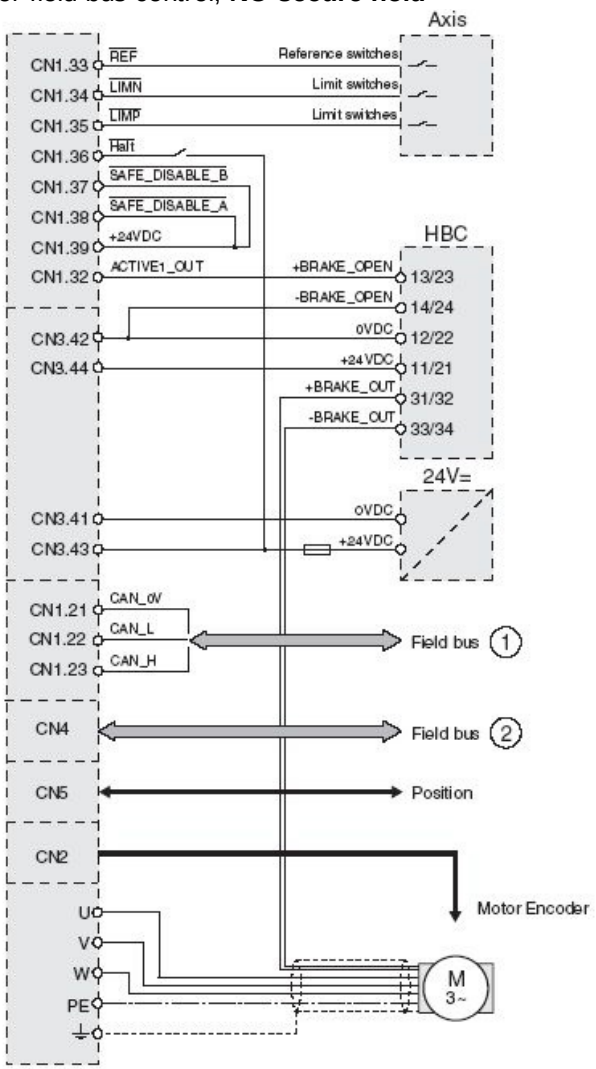
LXM05AD10M2

Terminals 33-39 must be connected.

The motor brake (if present) must be connected via a holding brake control (HBC)

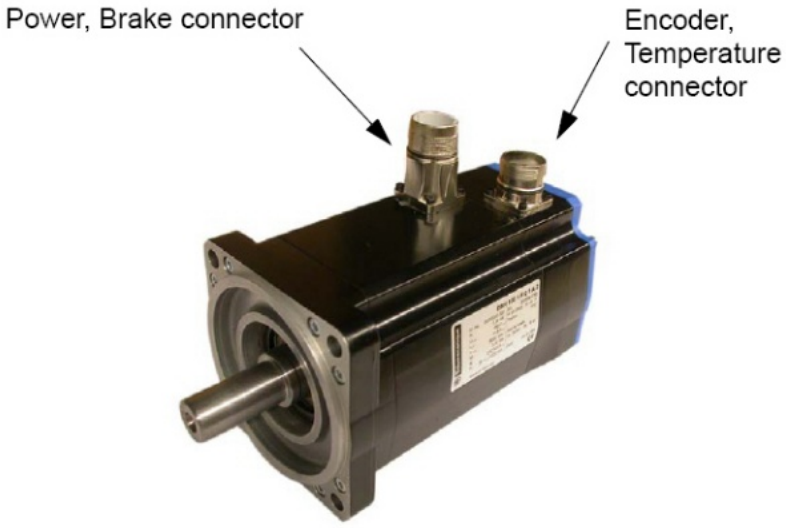
Use the RJ45 terminal for CANopen.

Wiring for field bus control, **NO secure hold**



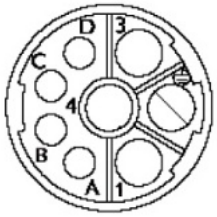
Servomotor

BSH0702P02A2A



Power Cable 3 m

VW3 M5 101R30



Pin	Designation (lead no.)	Meaning	Range
1	U	power	3 AC 0 - 480V
2	PE	shield	
3	W	power	3 AC 0 - 480V
4	V	power	3 AC 0 - 480V
A	brake +	brake	DC 24V
B	brake -	brake	DC 0V
C	-	not assigned	
D	-	not assigned	

Feedback Cable 3 m

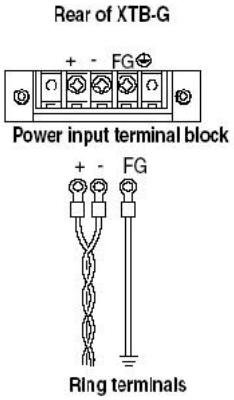
VW3 M8 101R30

Pin	Designation (lead no.)	Meaning	Range
1	Sensor PTC	Temperature	
2	Sensor PTC	Temperature	
3	-	not assigned	
4	REF SIN	REF signal	
5	REF COS	REF signal	
6	Data +	RS 485	
7	Data -	RS 485	
8	SIN +		
9	COS +		
10	U	power	DC 7 - 12V
11	GND	Ground	DC 0V
12	-	not assigned	

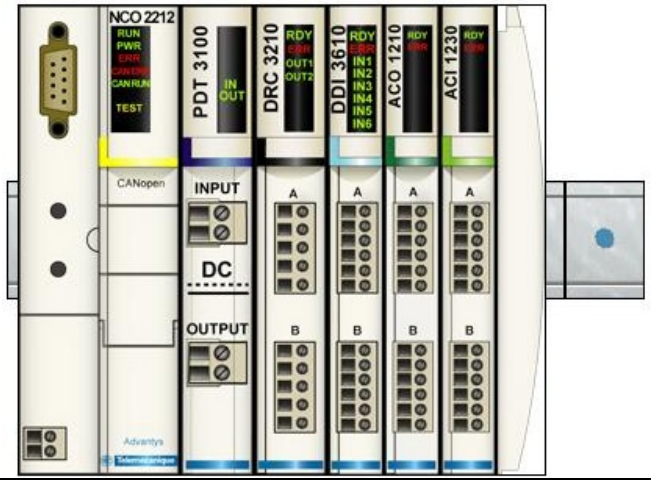
HMI

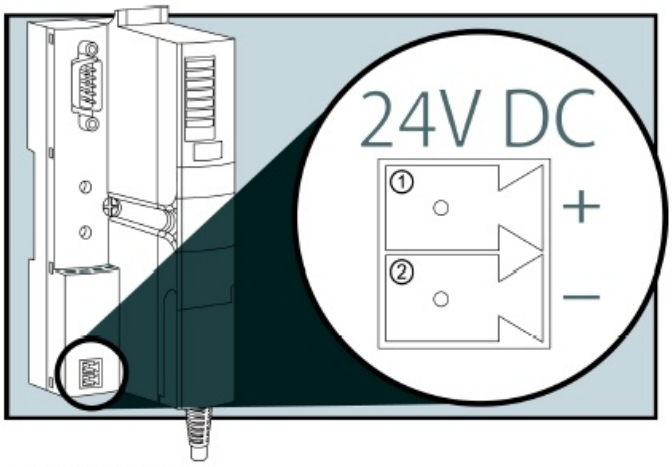
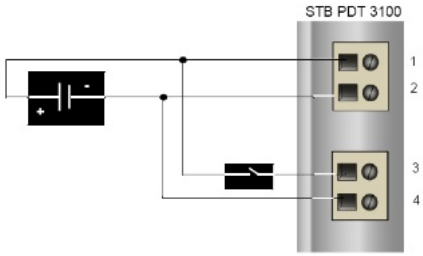
Magelis

XBT-G2330



Advantys STB



Advantys STB **CANopen Bus adapter** **STB NCO 2212**	 1 receptacle 1—24 VDC 2 receptacle 2—common voltage
Advantys STB **Field Power supply** **STB PDT 3100** Note: the power supply to the inputs and outputs can be individually switched (e.g. for E-OFF)	 1 +24 VDC sensor bus power 2 –24 VDC return line of sensor power supply 3 +24 VDC actuator bus power 4 –24 VDC return line of actuator power supply

Software

General

The major software components involve the programming of the Twido PLC, including its CANopen configuration, and the creation of the HMI displays.

Twidosoft is used to program the PLC.

Vijeo Designer is used to create the HMI application for the Magelis XBT-G 2330 display.

The Advantys configuration software is used to configure the Advantys STB I/O platform.

Although the Lexium 05 and Altivar 31 can be configured using their front panels, for more comfort Powersuite is used. Powersuite not only facilitates the configuration of the devices but also offers the possibility of archiving and documenting the configuration. Archiving offers considerable advantages for maintenance and updating. The software can also be used to optimise the parameters online.

In order to use the software your PC must have one of the following Microsoft Windows® operating systems:

- Windows® 2000 *or*
- Windows® XP

The software is installed on your PC under the following default paths:

- Twidosoft
  C:\Programme\Schneider Electric\TwidoSoft

- Vijeo-Designer
  C:\Programme\Schneider Electric\VijeoDesigner

- Advantys Configuration Software
  C:\Programme\Schneider Electric\Advantys

- PowerSuite
  C:\Programme\Schneider Electric\PowerSuite

Communication

General

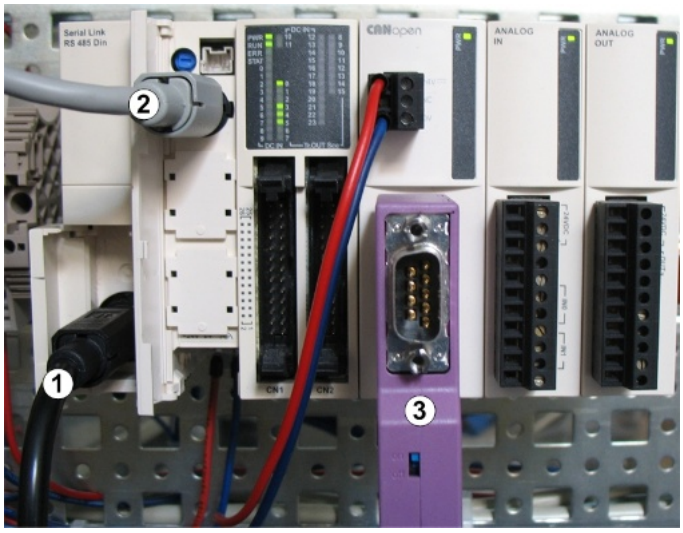
The buses below are used for communication between devices:

- CANopen
- Modbus

CANopen is used for communication at field bus level between the Twido PLC, the Lexium 05 and Altivar 31 drive controllers and the remote Advantys I/O platform.

Should the PLC be reset, e.g. using “Reset Communication”, the parameters are reset to the factory values. These must be re-initialised using SDOs via CANopen.

Modbus is used between the Magelis graphic touch panel (HMI) and the Twido PLC.

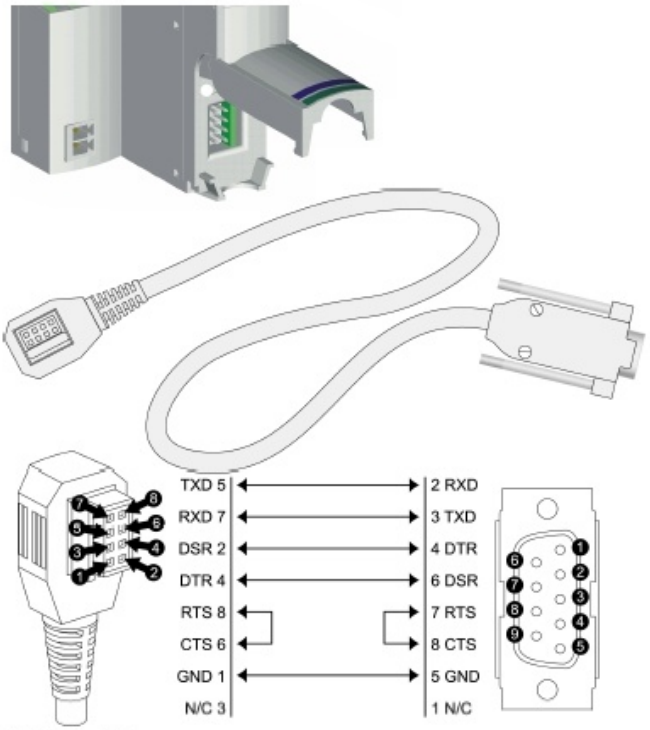
Twido PLC The following module extensions and cables are required**:** **1. TWD NOZ 485D** RS485 Expansion To connect the HMI with the PLC use cable **XBTZ968** **2. TWD LMDA 40DTK** Twido-SPS Use cable **TSX PCX 1031** to make the serial connection from the PC to the PLC. Use Twidosoft to program the PLC **3. TWD NC01M** **CANopen extension** The standard CANopen plugs and cables are used.	

HMI **Magelis** **XBT-G2330**	
XBTZ968 **XBTZG999** Communication cables for PLC including adapter.	
XBTZG915 Serial communication cable to PC (with Vijeo Designer). The Ethernet interface can be used as an alternative.	

Advantys STB

Programming cable STB XCA 4002

For connection to the serial interface of a PC with Advantys software.



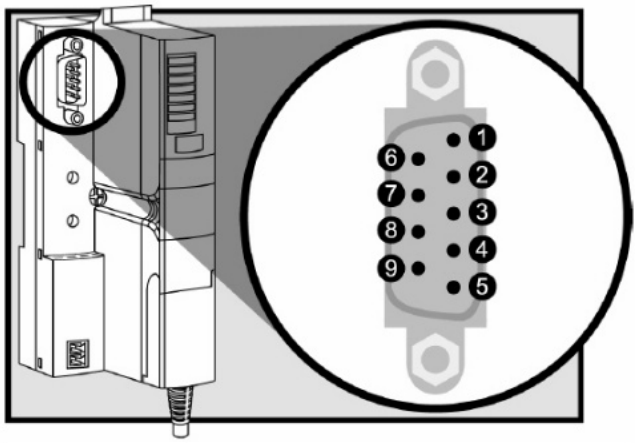
TXD transmit data
RXD receive data
DSR data set ready
DTR data terminal ready
RTS request to send
CTS clear to send
GND ground reference
N/C not connected

Advantys STB

CANopen – Bus adapter

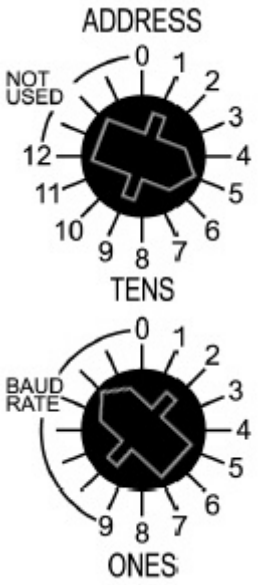
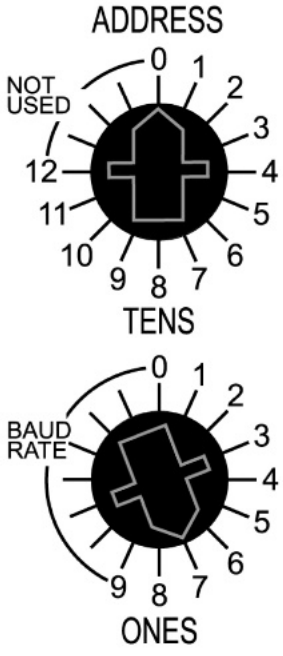
STB NCO 2212

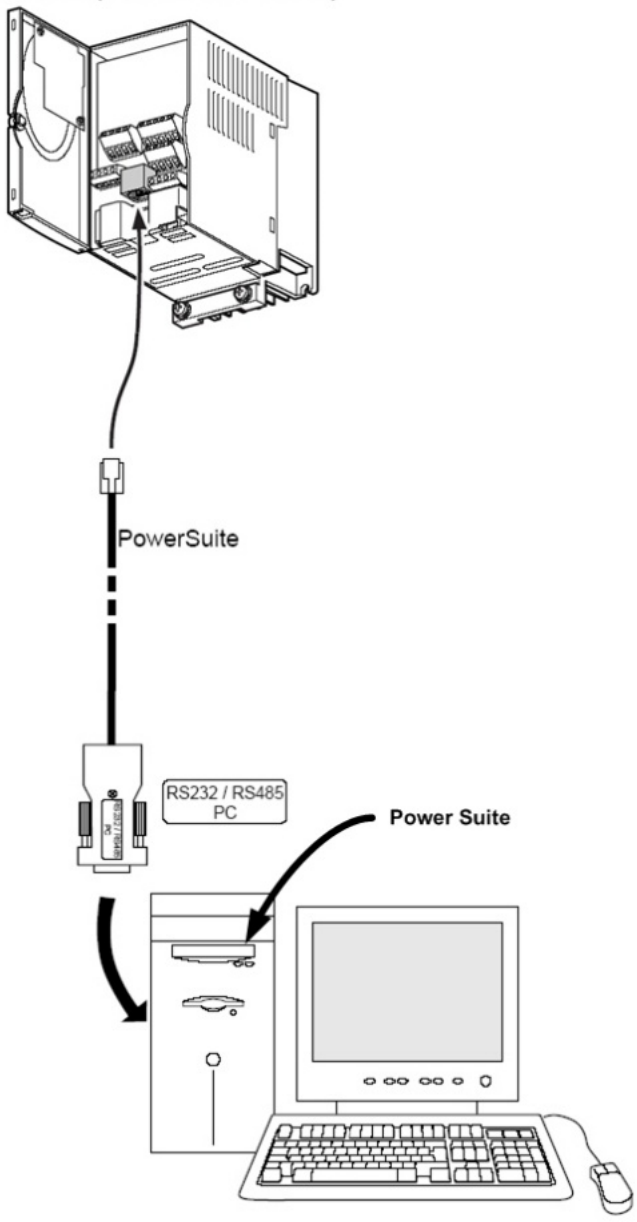
CANopen Fieldbus connection



Pin	Signal	Beschreibung
1	Nicht verwendet	Reserviert
2	CAN_L	CAN Busleitung, Low
3	CAN_GND	CAN-Masse
4	Nicht verwendet	Reserviert
5	CAN_SHLD	Optionale CAN-Abschirmung
6	GND	Optionale Masse
7	CAN_H	CAN-Busleitung, High
8	Nicht verwendet	Reserviert
9	Nicht verwendet	Reserviert

Hinweis: Die Pinnummern entsprechen der Darstellung in der Abbildung oben.

Advantys STB **CANopen bus adapter** **STB NCO 2212** **CANopen baud rate** 	The rotary switches on the STB NCO 2212 CANopen NIM are used to set the network node address and the Advantys STB island's baud rate. 1. Disconnect the island's power supply. 2. Set the lower rotary switch to any position between 9 and 0 (baud rate setting is marked). 3. Select the baud rate you wish to use for fieldbus communication. Select an appropriate baud-rate setting for your system and network requirements. 0 - 10,000 bps 4 - 250,000 bps 1 - 20,000 bps 5 - 500,000 bps 2 - 50,000 bps 6 - 800,000 bps 3 - 125,000 bps 7 - 1 Mbps In this example we have selected setting **5** (500,000 bps). 4. Turn the upper rotary switch to the position corresponding to the baud rate you have selected (e.g., **5**). 5. Power up your island to use the new settings. The NIM only reads the rotary-switch settings on power-up.
Advantys STB **CANopen bus adapter** **STB NCO 2212** **CANopen address** 	The rotary switches on the STB NCO 2212 CANopen NIM are used to set the network node address and the Advantys STB island's baud rate. 1. Be sure to set the required baud rate (following the procedure described above) before setting the network node address. 2. Disconnect the island's power supply. 3. Select a network node address that is available in your fieldbus network. 4. Set the lower rotary switch to the position corresponding to the units of the required node address. For network node address **7**, the lower switch would be set to **7**. 5. Set the upper rotary switch to the position corresponding to the ten's and hundred's of the required node address. For network node address **7**, the upper switch would be set to **0**. 6. Switch on Advantys STB. The NIM only reads the rotary-switch settings on power-up.

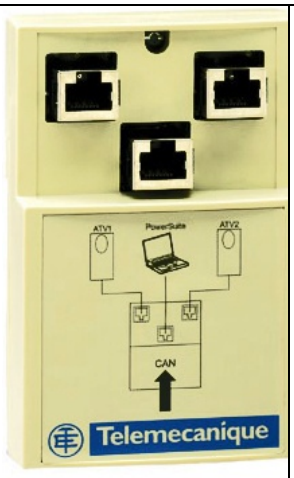
Connection cable set for PowerSuite **VW3 A8106** You will need the RS232 to RS485 adapter and the PowerSuite cable for the connection between the PC and the PowerSuite software and VSDs or servos.	ATV 31 (PowerSuite ⩾ V2.0.0) LXM 05 (PowerSuite ⩾ V2.2.0) 
CANopen junction box **VW3 CAN TAP2**	

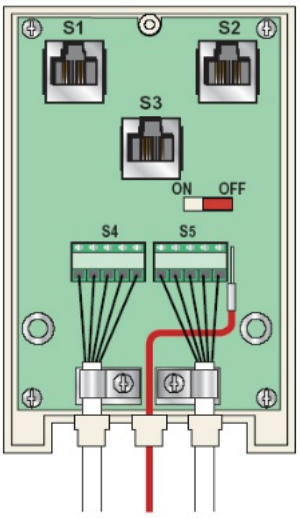
CANopen junction box

VW3 CAN TAP2

For this application example, the slide switch must be set to OFF.

If, unlike in this application, the junction box does not have an outgoing CANopen bus, the line terminator must be activated (i.e., set to ON).





Pin	Signal	Wire colour	Description
1	GND	Black	Ground
2	CAN_L	Blue	CAN_L bus line
3	SHLD	(bare cableshield)	Optional shield
4	CAN_H	White	CAN_H bus line
5	(V+)	Red	Optional supply

CANopen pre-assembled connection cable

VW3 CAN CARRxx

This cable is used to connect the junction box to the ATV31 and LXM05.

VW3 CAN CARR1
(length: 1.0 m)

VW3 CAN CARR03
(length: 0.3 m)

CANopen connector

VW3 CAN KCDF 90TP
or
VW3 CAN KCDF 90T

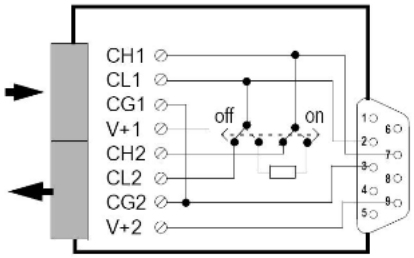
This connector (with terminal resistor) is used for the link to the Advantys STB I/O island.

It is advisable to have at least one connector with a CANopen analyser connection

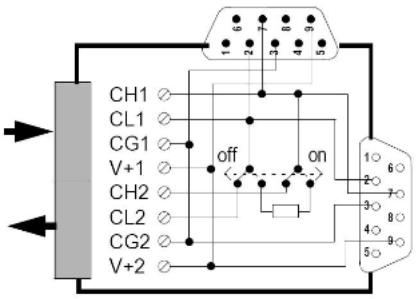
At the **end of the bus**, the terminating resistor must be activated. To do this, set the switch to ON and the bus cable must be connected to the incoming side.

CANopen connector assignment

VW3 CAN KCDF xxxx



TSXCANKCDF90T
TSXCANKCDF180T



TSXCANKCDF90TP

Signal	Terminal block 1	Terminal block 2	Wire color
CAN_H	CH1	CH2	white
CAN_L	CL1	CL2	blue
CAN_GND	CG1	CG2	black
CAN_V+	V+1	V+2	red

CANopen cable

TCX CAN Cx yy

The cable is available in various versions (x):

Standard
No Flame
Heavy Duty

and various lengths (yy):

50,100, 300 m.

Implementation

Introduction

This chapter describes how to initialize, parameterize, program, and start up the system.

Function

Switching on and Functional description

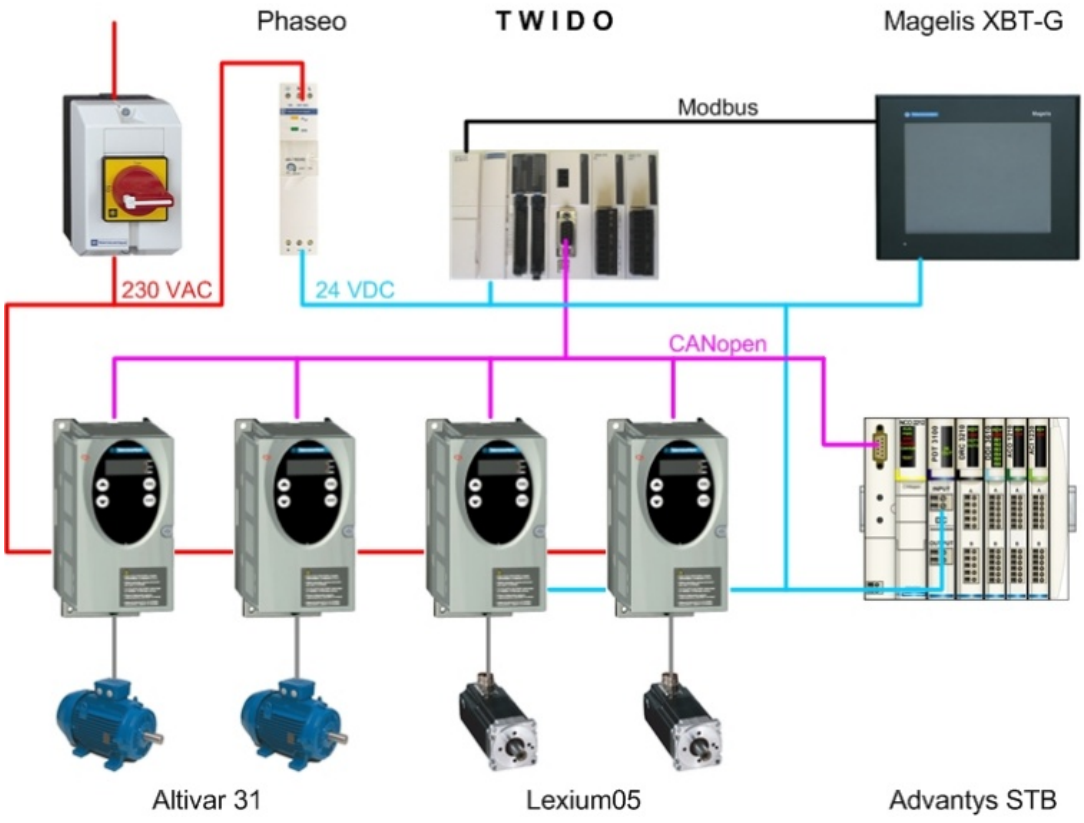
1. After the power is switched on, all devices run through the initialization stage and the PLC starts the communication. The status bar displays system information. In this application, Manual mode (via HMI) is implemented

2. On the BUS screen, the status of the CANopen bus can be viewed. Use the buttons **RESET NODE** and **TO OPERAT** to reset and restart the communications.

3. The drives can be controlled using the appropriate display screens. Use the button **RAMP** to start an automatic speed cycle for all the drives

4. If an error occurs, the error number is displayed on the ALARM screen. The user can look up the description of the error in the operating manual.

Layout



Communication

Introduction

This chapter describes the data passed via the communications bus (e.g. Modbus Plus or TCP/IP) that is not bound directly with digital or analog hardware.

The list contains:

- The device links
- Direction of data flow
- symbolic name and
- Bus address of the device concerned.

Device Links

The Modbus and CANopen bus systems are used in this application.

The following devices are networked via CANopen:

- A Twido PLC, bus address 127 (fixed,master)
- Two Lexium05 servo drives, bus addresses 3 and 4
- two Altivar 31 variable speed drives, bus addresses 5 and 6
- One Advantys STB I/O island, bus address 7

Two devices are connected via Modbus:

- Magelis panel XBT-G, bus address 1
- Twido PLC, bus addresses 2

CANopen Addresses

Twido-PLC is *CANopen-Master*	
Device	**CANopen Address**
1. Lexium 05	3
2. Lexium 05	4
1. Altivar 31	5
2. Altivar 31	6
Advantys STB	7

The Baud rate is 500 kb/sec.

Heartbeat is used to monitor the devices

CANopen COB-ID

Data Direction Device --> PLC (TPDO)						
Device	**1.PDO**	**2.PDO**	**3.PDO**	**4.PDO**	**5.PDO**	**6.PDO**
1. LXM05	183	---	---	483		
2. LXM05	184	---	---	484		
1. ATV31	---					694
2. ATV31	---					695
Advantys STB	187	287				
Data Direction PLC --> Device (RPDO)						
Device	**1.PDO**	**2.PDO**	**3.PDO**	**4.PDO**	**5.PDO**	**6.PDO**
1. LXM05	203	---	---	503		
2. LXM05	204	---	---	504		
1. ATV31	---					684
2. ATV31	---					685
Advantys STB	207	208				

Data Link PLC <> LXM05

Twido *(CANopen-Master)*		Lexium 05 *(CANopen-Slave)*	
Data Direction PLC <-- LXM (TPDO)			
1.LXM	**2.LXM**	**Index**	**Designation**
%IWC1.0.0	%IWC1.2.0	6041	Drivecom status register
%IWCD1.1.0	%IWCD1.3.0	606B	Velocity actual value
%IWCD1.1.2	%IWCD1.3.2	6064	Position actual value
Data Direction PLC --> LXM (RPDO)			
1.LXM	**2.LXM**	**Index**	**Designation**
%QWC1.0.0	%QWC1.2.0	6040	Drivecom command reg.
%QWCD1.1.0	%QWCD1.3.0	60FF	Target velocity
%QWCD1.1.2	%QWCD1.3.2	607A	Target position

Data Link PLC <> ATV31

Twido *(CANopen-Master)*		Altivar 31 *(CANopen-Slave)*	
Data Direction PLC <-- ATV (TPDO)			
1.ATV	**2.ATV**	**Index**	**Designation**
%IWC1.4.0	%IWC1.5.0	6041	Drivecom status register
%IWC1.4.1	%IWC1.5.1	6044	Velocity actual value
%IWC1.4.2	%IWC1.5.2	603F	Error Code
Data Direction PLC --> ATV (RPDO)			
1.ATV	**2.ATV**	**Index**	**Designation**
%QWC1.5.0	%QWC1.5.0	6040	Drivecom command reg.
%QWC1.5.1	%QWC1.5.1	60FF	Target velocity

Data Link PLC <> STB

Twido *(CANopen-Master)*		Advantys STB *(CANopen-Slave*
Data Direction PLC <-- STB (TPDO)		
Address	**Name**	**Designation**
%IWC1.6.0	STB IN DI1	Digital Input Block
%IWC1.6.1	STB IN DI2	Digital Input Block
%IWC1.6.2	STB IN DI3	Digital Input Block
%IWC1.6.3	STB IN DI4	Digital Input Block
%IWC1.6.4	STB IN DI5	Digital Input Block
%IWC1.6.5	STB IN DI6	Digital Input Block
%IWC1.7.0	STB IN AI1	Analog Input Block
%IWC1.7.1	STB IN AI2	Analog Input Block
Data Direction PLC --> STB (RPDO)		
Address	**Name**	**Designation**
%QWC1.6.0	STB OUT DO1	Digital Output Block
%QWC1.7.0	STB OUT AO1	Analog Output Block
%QWC1.7.1	STB OUT AO2	Analog Output Block

General Adressing

The PLC example program uses different hardware, discretes and memory words. The following list contains keys to aid the understanding of the addressing system used.

Type	Address	Comment
Digital Inputs	%I0.0 ... %I0.23	TWD_IN_DI00 ... TWD_IN_DI023
Digital Outputs	%Q0.0 ... %Q0.15	TWD_OUT_DO00 ... TWD_OUT_DO15
Analog Inputs	%IW2.0 ... %IW2.1	TWD_IN_AI1 ... TWD_IN_AI2
Analog Outputs	%QW3.0	TWD_OUT_AO1
Internal Word	%MW 0 ... 3000	max. possible address
Constant Word	%KW 0 ... 256	max. possible address
Internal Doubleword	%MD 0 ... 2999	max. possible address
Constant Doubleword	%KD 0 ... 255	max. possible address
Internal Floating Point	%MF 0 ... 2999	max. possible address
Constant Floating Point	%KF 0 ... 255	max. possible address
CANopen Inputs	%IWC...	see table above
CANopen Outputs	%QWC...	see table above

CANopen Function

CAN_CMD

Twidosoft provides the function CAN_CMD for access to the CANopen Interface. This function allows control of the CANopen bus (e.g. Reset Node) and transfer of SDOs to a specific device on the network. The parameters for the commands are forwarded using memory words (%MWx). The following table describes what action is taken when the parameters are given a particular value (%MWx to %MWx+5).

%MWx	%MWx+1		%MWx+2		%MWx+3		%MWx+4		%MWx+5		Action
	Bit [0-7]	Bit [8-15]	Bit [0-7]	Bit [8-15]	Bit [0-7]	Bit [8-15]	Bit [0-7]	Bit [8-15]	Bit [0-7]	Bit [8-15]	
1	0		—								Reset CANopen communication.
1	1										Reset CANopen nodes.
2	0										Switch from operational to pre operational mode.
2	1										Switch to operational mode.
3 or 4											3 => Start Read SDO command. 4 => Start Write SDO command.
	Node										Node = 1-16 => Node address
			Index								PDO object index.
					Len	Sub					Sub = 0-255 => Object sub-index Len = Length of data in byte
							Data 1				Payload according to the length field (Len) value
									Data 2		Payload according to the length field (Len) value

CANopen CAN_CMD

SDOs

The following table lists the SDOs used in the application. Only one SDO is active at any one time. The order of invocation depends on the active devices on the bus and is managed in the PLC.

The initial values are stored as constants (%KW) and are copied to memory (%MW) at program start-up.

CANopen SDO Communication		r/w	SDO	Memory		Initial Value in Word x					
				from %KW	from %MW	1.	2.	3.	4.	5.	6.
General	Spare SDO e.g..	r	1.	---	409						
	Spare SDO e.g..	w	2.	---	415						
	Init to PreOp (Reset Communication) to Init mode	s	3.	21	421	1	0	---	---	---	---
	(Reset Node)	s	4.	27	427	1	1	---	---	---	---
	Operational to PreOp	s	5.	33	433	2	0	---	---	---	---
	to Operational mode	s	6.	39	439	2	1	---	---	---	---
STB	Status Analog value	r	7.	45	445	3	7	6423	00 02	---	
	Status Analog value	w	8.	51	451	4	7	6423	00 01	1	
	Inhibit	r	9.	57	457	3	7	1801	03 02		
	Inhibit	w	10.	63	463	4	7	1802	03 02	2000	
1. LXM	Inhibit	r	11.	69	469	3	3	1803	03 02		
	Inhibit	w	12.	75	475	4	3	1803	03 02	2000	
	Ramp Start (ACC)	r	13.	81	481	3	3	6083		---	
	Ramp Start (ACC)	w	14.	87	487	4	3	6083	00 04	2000	
	Ramp Stop (DEC)	r	15.	93	493	3	3	6084		---	
	Ramp Stop (DEC)	w	16.	99	499	4	3	6084	00 04	2000	
	Mode	r	17.	105	505	3	3	6060	00 02	---	
	Mode	w	18.	111	511	4	3	6060	00 01		
	Error	r	19.	117	517	3	3	603F	00 02	---	
2. LXM	Inhibit	r	20.	123	523	3	4	1803	03 02		
	Inhibit	w	21.	129	529	4	4	1803	03 02	2000	
	Ramp Start (ACC)	r	22.	135	535	3	4	6083		---	
	Ramp Start (ACC)	w	23.	141	541	4	4	6083	00 04	2000	
	Ramp Stop (DEC)	r	24.	147	547	3	4	6084		---	
	Ramp Stop (DEC)	w	25.	153	553	4	4	6084	00 04	2000	
	Mode	r	26.	159	559	3	4	6060	00 02	---	
	Mode	w	27.	165	565	4	4	6060	00 01		
	Error	r	28.	171	571	3	4	603F	00 02	---	

s - System
r - read
w - write

HMI

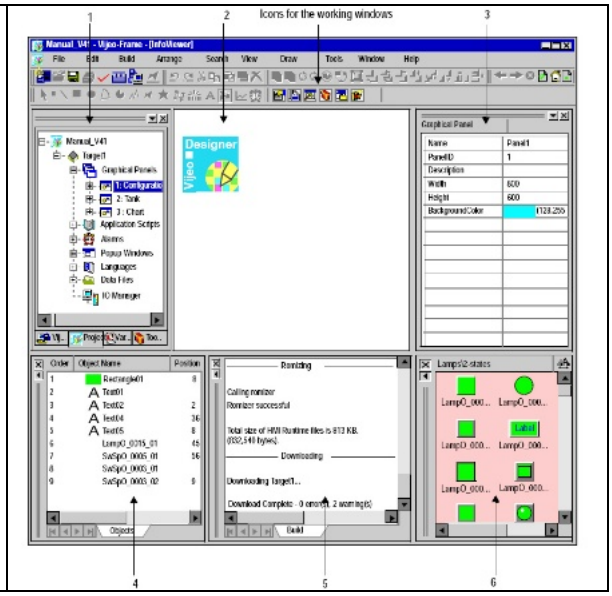
Introduction

This section describes how to set up the screens for the Magelis HMI. Vijeo Designer is the software used.

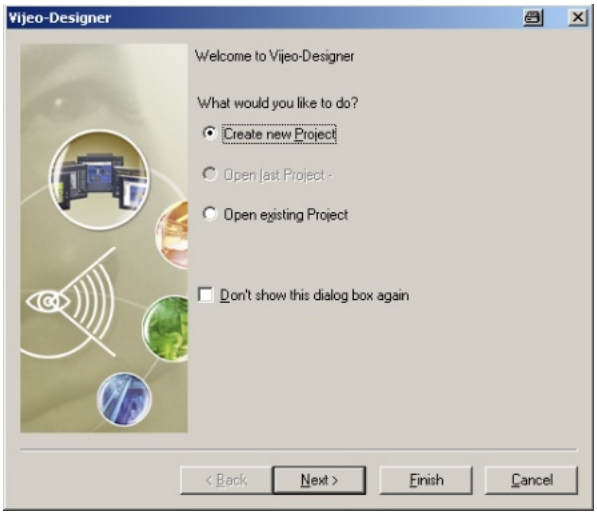
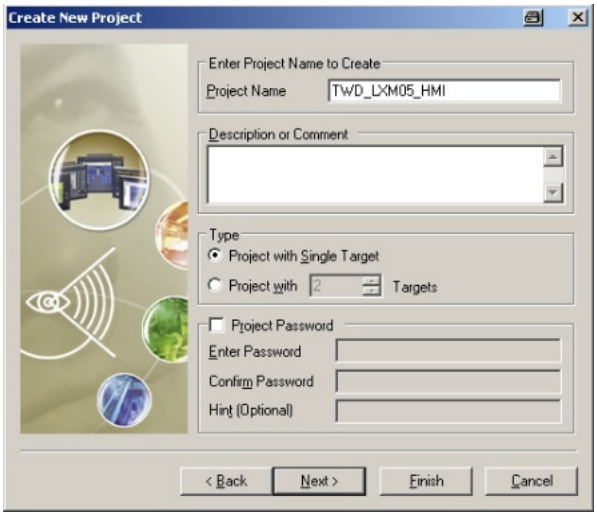
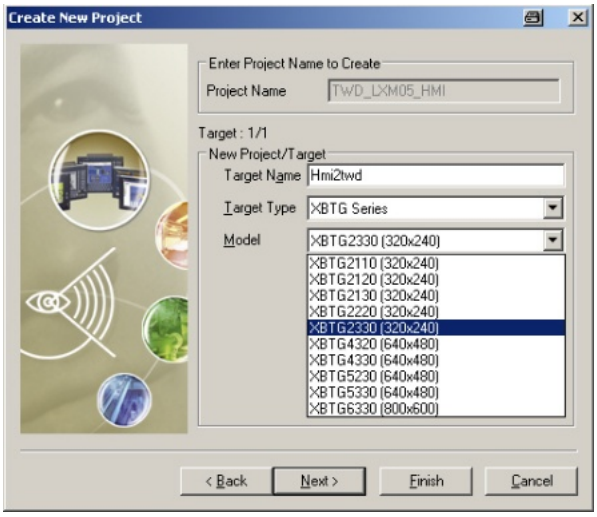
Proceed as follows to integrate the HMI:

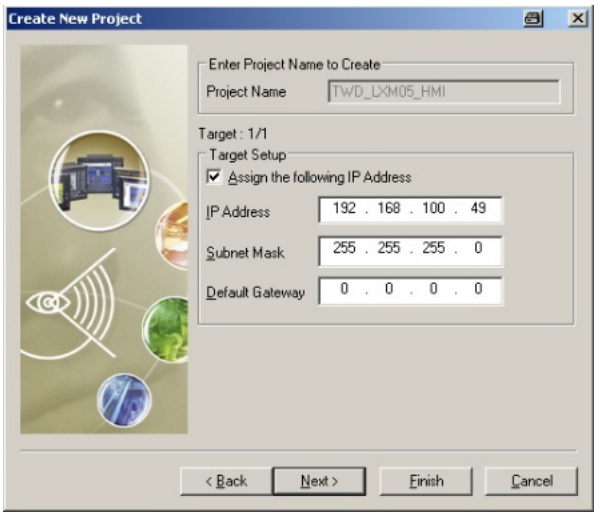
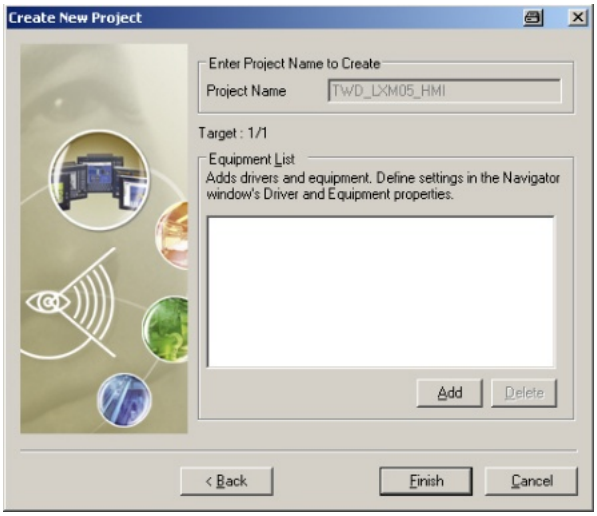
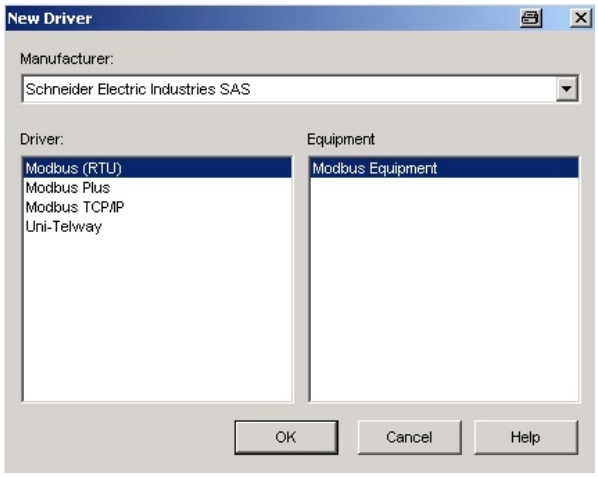
- Create a new project
- Set Up the communication
- Create variables
- Create a screen
- CANopen Status display
- Build and Download Project
- Application summary

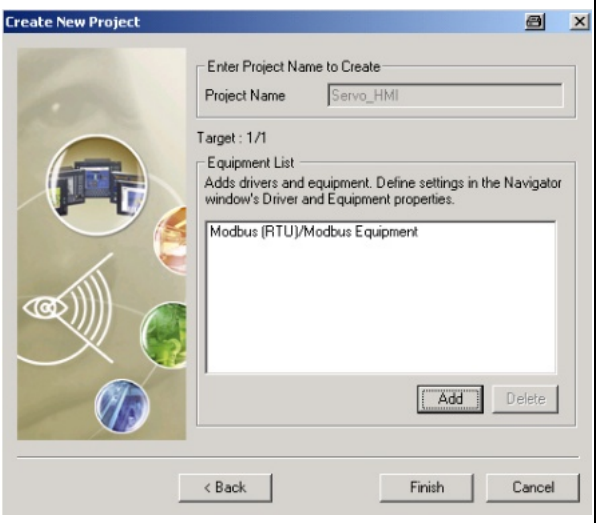
Vijeo Designer environment

1	The Vijeo Designer environment consists of: 1. Navigator 2. Information display 3. Inspector 4. Data list 5. Feedback area 6. Toolbox	

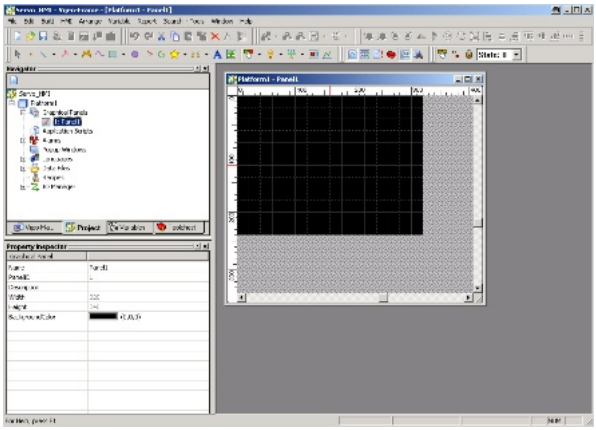
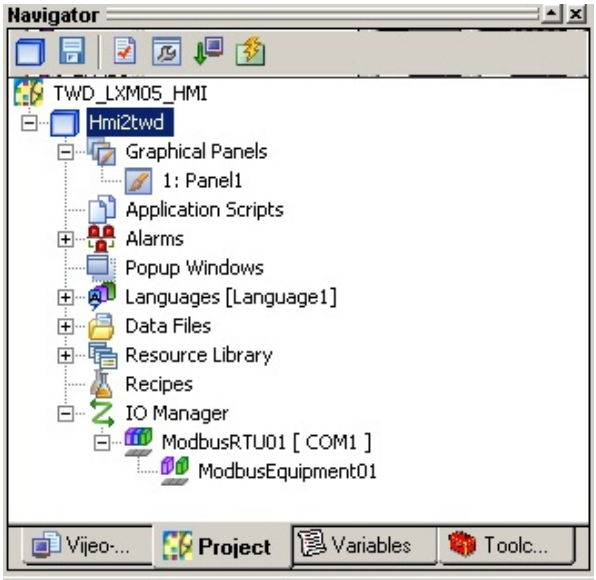
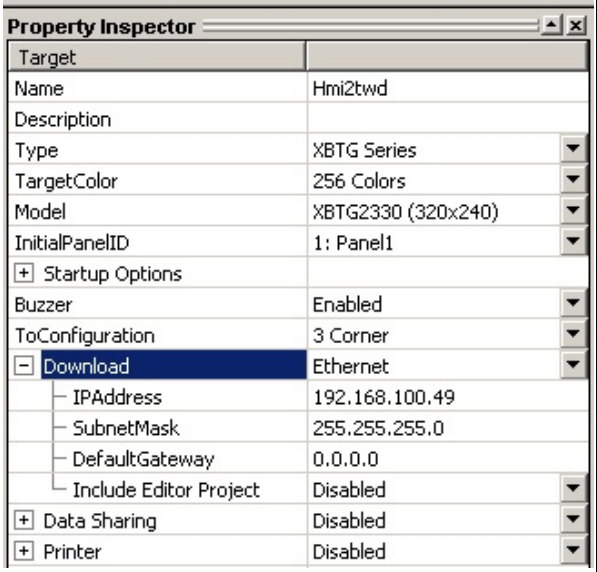
Create a new project

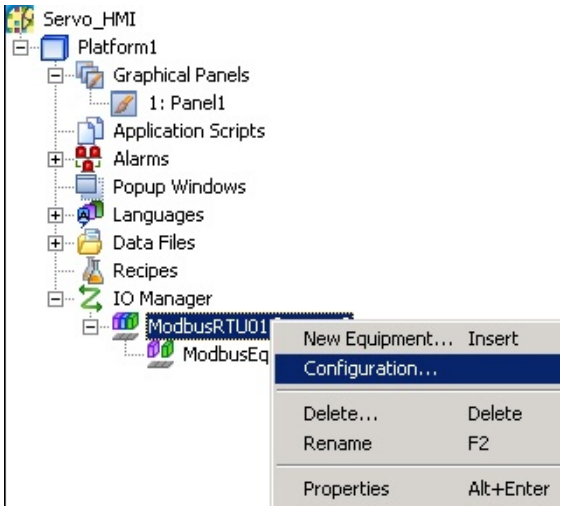
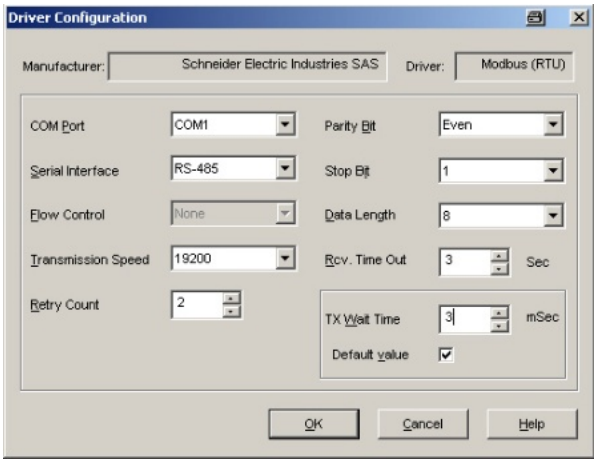
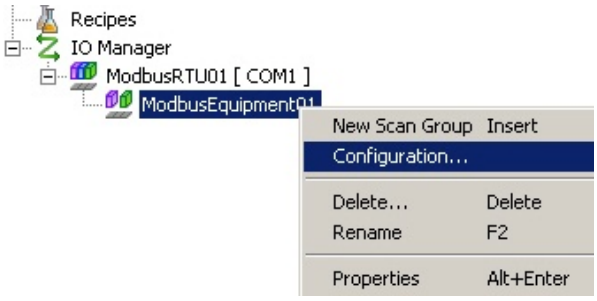
1	Start up Vijeo designer and select **Create new project**. Continue with **Next>**.	
2	Assign a **project name** and, if required, enter a **description or comment.** Continue with **Next>**.	
3	Select the target device used: **Target name**: **Hmi2twd** **Target type**: **XBTG Series** **Model**: **XBTG2330** Continue with **Next>**.	

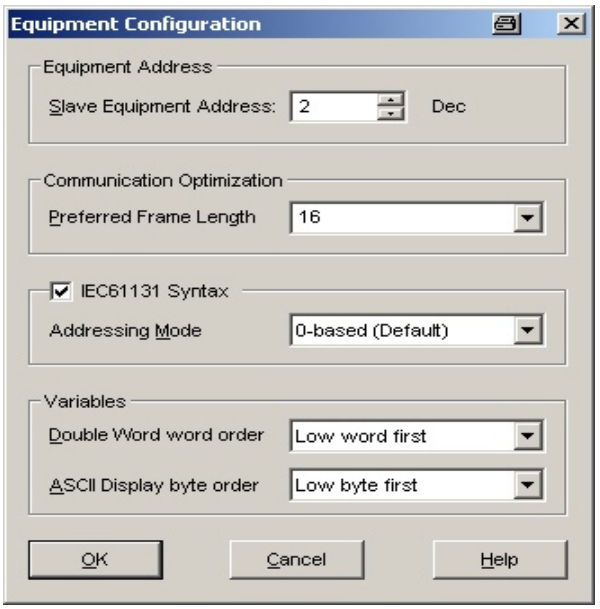
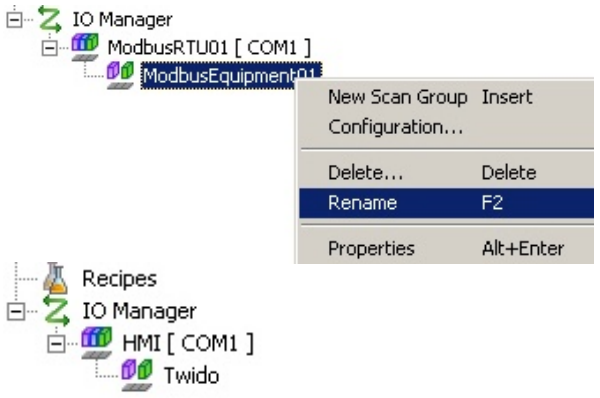
4	To use the Ethernet interface you must enter an Ethernet address for the target device. If required, enter **IP address** **Subnet Mask** **Default Gateway** Continue with **Next>**.	
5	To exchange data with other devices the Magelis HMI needs a communications driver. Attach the new driver by clicking on **Add.**	
6	In the **New Driver** dialog select: **Manufacturer: Schneider Electric Industries SAS** **Driver: Modbus (RTU)** **Equipment: Modbus Equipment** Exit with **OK** to accept the driver.	

7	The new driver is attached. Exit with **Finish** to complete the project creation.	

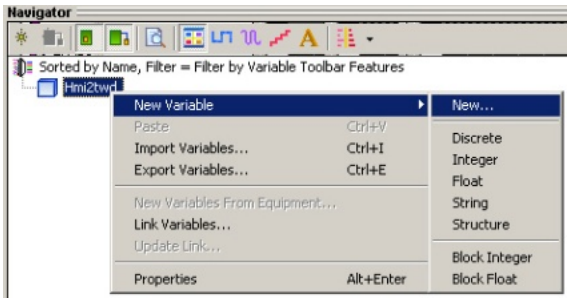
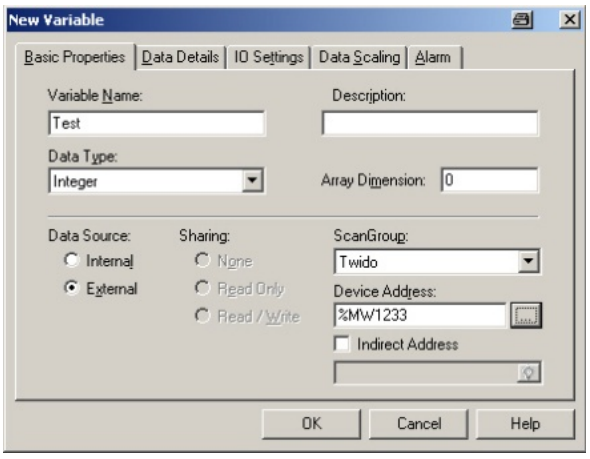
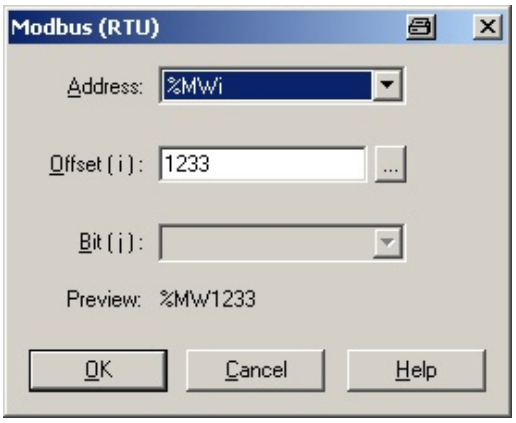
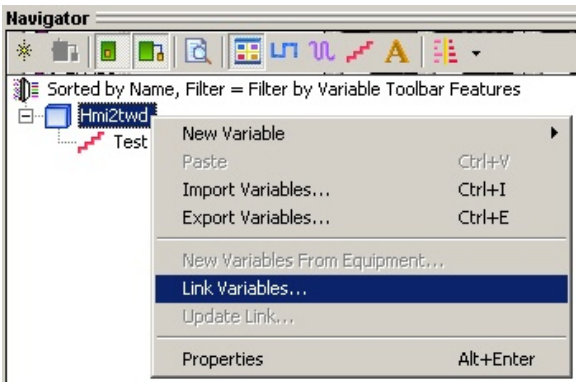
Set Up the Communication

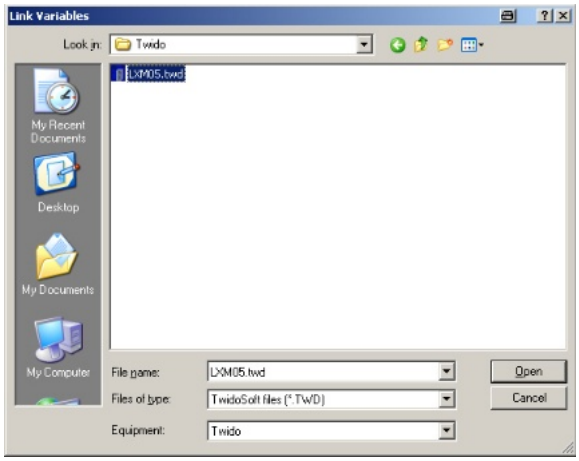
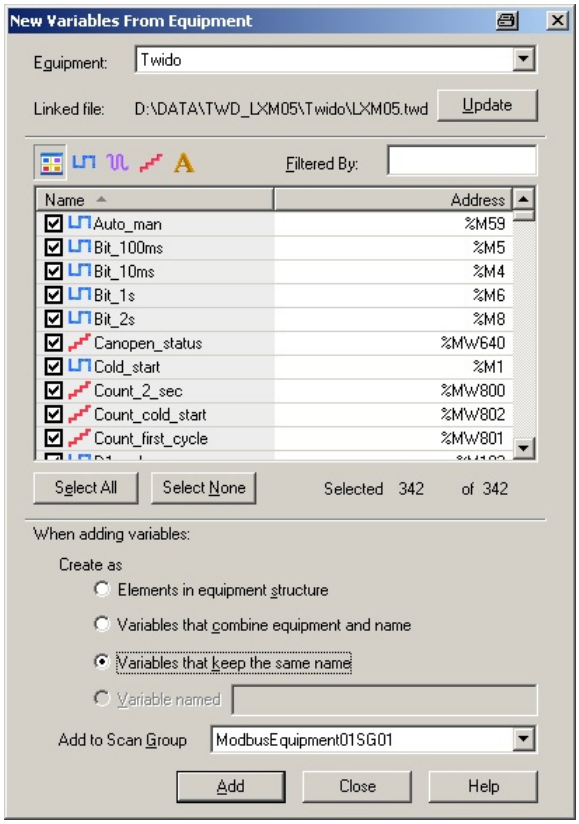
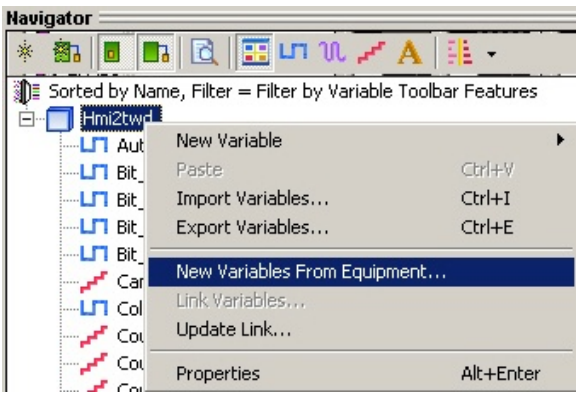
1	Once the project has been created Vijeo Designer displays the working environment.	
2	Right mouse click on the platform name in the **Navigator** to view the project attributes in the **Property Inspector**. Select the **download** setting for the connection between PC and Magelis. Ensure that the setting shows **Ethernet** and that the **IPAddress** and **SubnetMask** are valid.	 

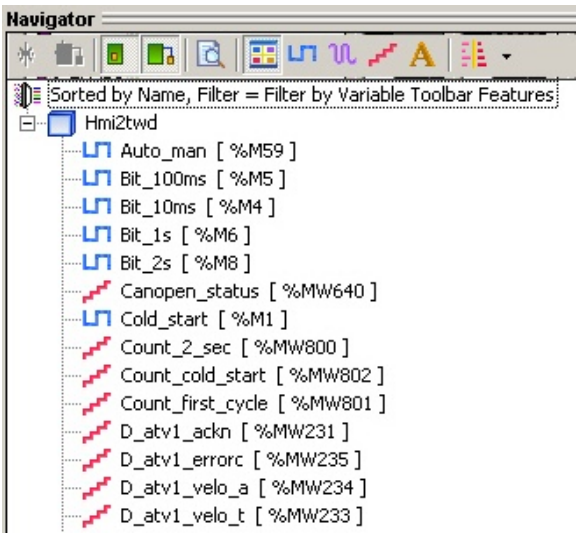
3	To configure the Modbus driver select: **I/O Manager** **-> ModbusRTU01** **->Configuration....** in the **Navigator.**	
4	Driver configuration: • COM1 (RS485) • Even • 8 data bits • 1 stop bit • 19200 baud The configuration must be the same as in the PLC. Enter the correct values and exit the dialog with **OK.**	
5	To configure the device, right mouse click on the connection in the **Navigator** and select: **Configuration..**	

6	In the **Equipment Configuration** dialog set the **slave equipment address** to **2.** The **Preferred Frame Length** is **16.** Click the box for **IEC61131 Syntax** and select **0-based(default)** so that the same **Addressing Mode** is used as in the PLC (%MWxxx etc.). Exit with **OK**.	
7	You can **Rename** the **Navigator** entries if you wish.	

Create variables

1	To create a variable, switch to the **Variable** tab in the **Navigator**. Right mouse click on the project name and select : **New Variable->New..** in the pop-up menu.	
2	In the **New Variable** dialog specify the variable properties: • **Variable Name** • **Data Type** • **Data Source – External** • **Scan Group** • **Address in the PLC**	
3	The variable address is any valid PLC address, e.g. bits %M, words %MW, double-words %MD, floating point %MF. Any values used in the visualisation of the process must be transferred to these data types in the PLC in order for the HMI to access them.	
4	Variables can also be imported or exported. another possibility to link directly to them in the PLC Select the **Variables** tab in the **Navigator.** Right mouse click on the project name in the **Variable** tab and select **Link Variables..** in the pop-up menu	

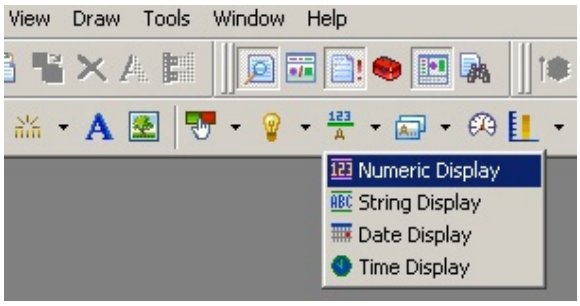
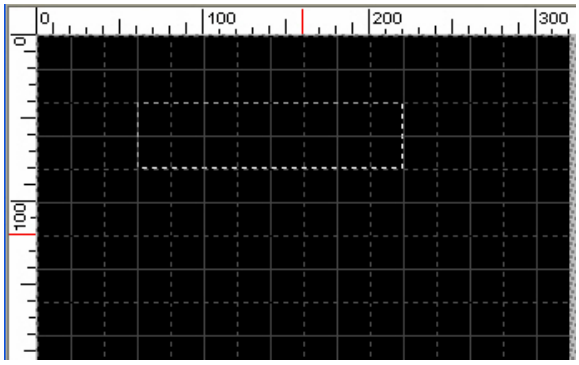
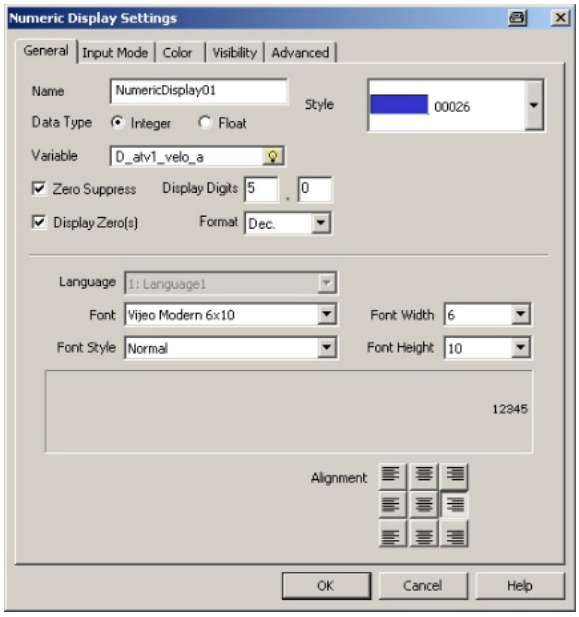
5	In the **Link Variables** window give the path to the file, the type of file (*.TWD) and the **Equipment** (here as named above **Twido**). Select the file and click on **Open.**	
6	Once the file was successfully opened you will be offered a selection list of the available variables Mark the variables you wish to link to by clicking on the check box for that variable To facilitate legibility the variable names in the PLC and HMI should be the same. Click on **Variables that keep the same name.** Once you have made your selection. Click on **Add** to link to the variables and then **Close** to close the dialog**.**	
7	If you wish to link to more variables later you can use the **New Variables from Equipment..** function.	

8	All variables can now be viewed in the **Navigator** on the **Variables** tab.	

Create a screen

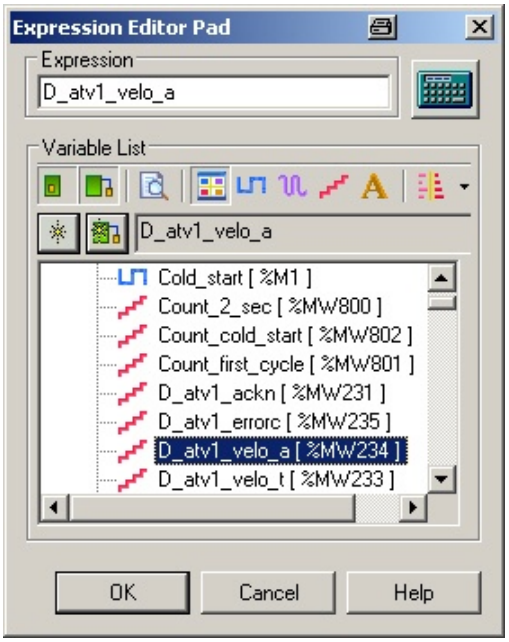
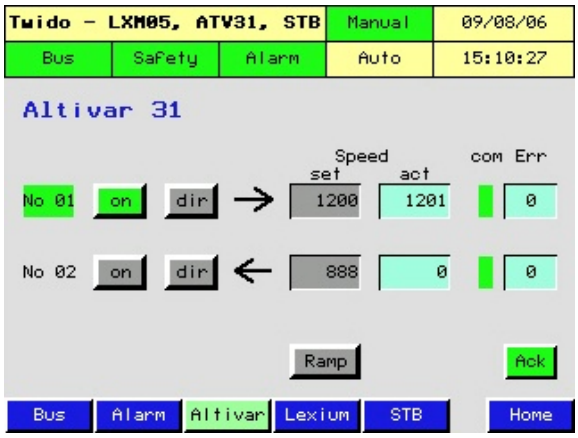
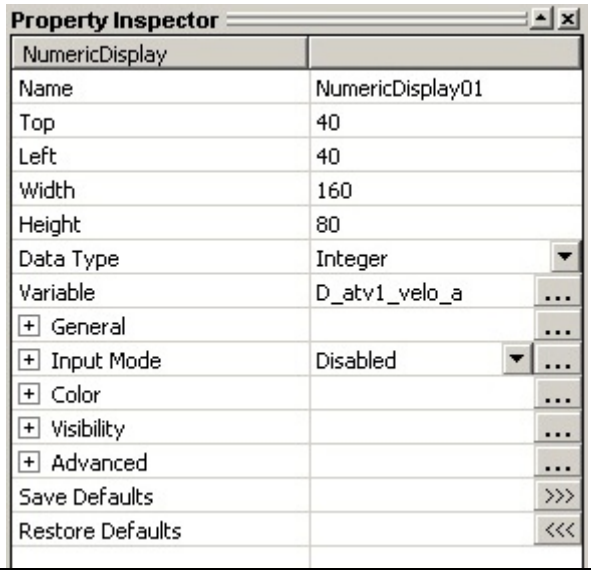
Example

As an example of how to create an animated screen, the creation of a numerical object is described here. The method is similar for creating other objects in the screen.

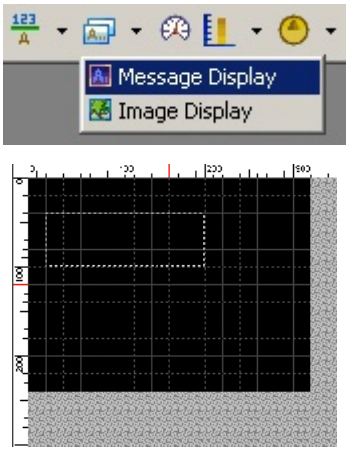
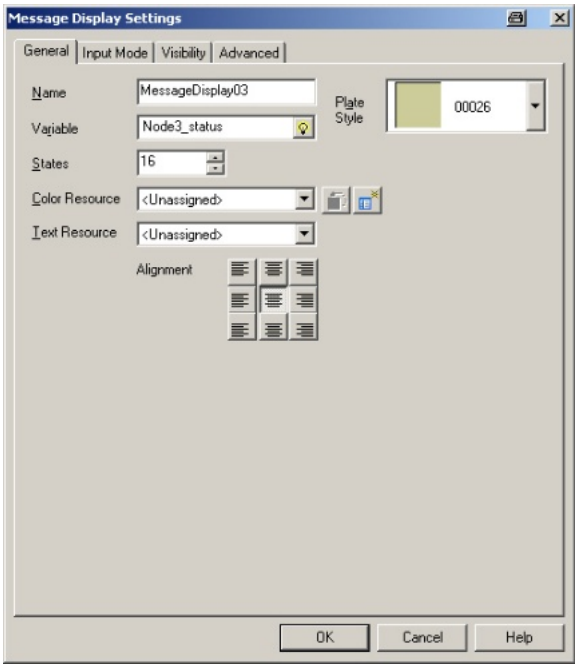
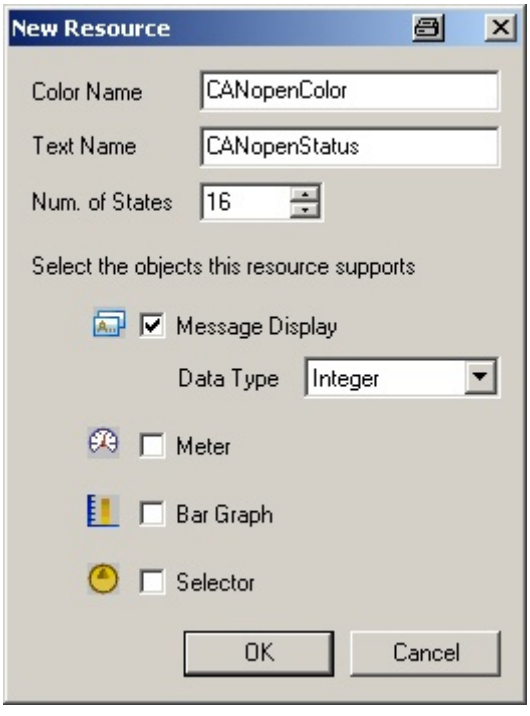
1	The different object types can be selected and inserted into the screen using the toolbar	
2	First mark the position and size of the object on the screen.	
3	Define the attributes for the object • **Name** • **Data Type** • **Variable** • **Style** • **Font Style** and size, The **Variable** can be edited manually or selected from a list by clicking on the **bulb** icon at end on the edit box. **Note:** if the given **variable** is undefined, it is indicated with red text.	

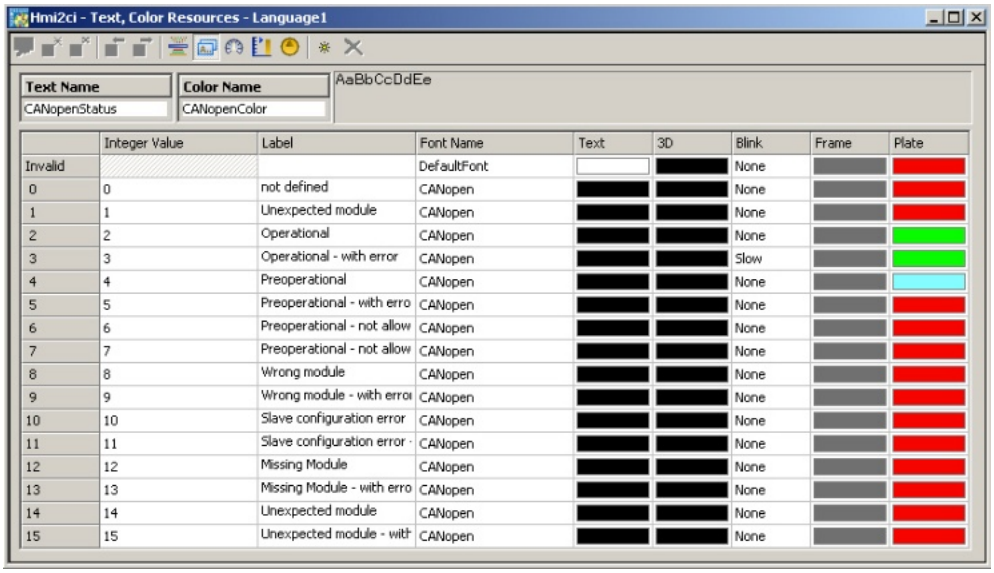
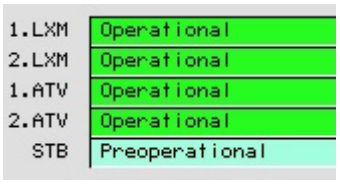
Continued on next page

Create a screen contd.

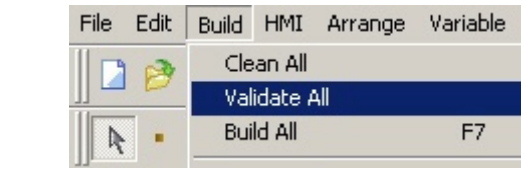
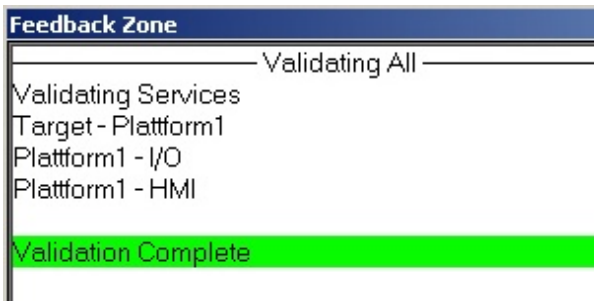
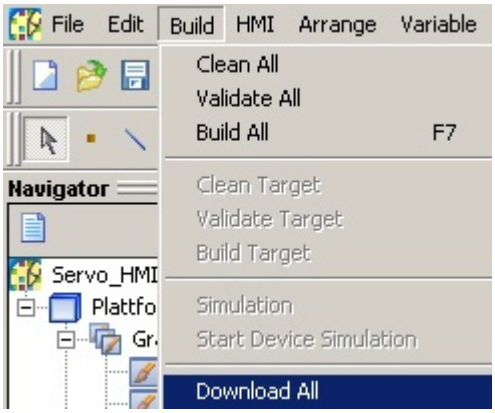
4	The variable to use for the animation can be selected from the list with a double click. Extra functions, e.g. the negation of the value of the variable, can be selected using the **calculator** icon in the toolbar.	
5	Here the finished screen with text fields, animated variables and control buttons.	
6	Every object on the screen has attributes that can be viewed in the **Property Inspector**. To view the attributes of an object, use right mouse click on the object.	

CANopen Status Display

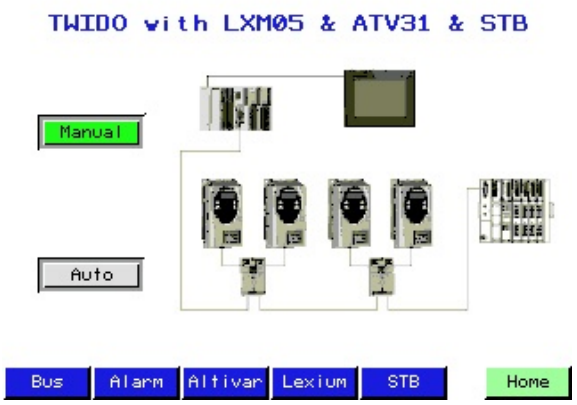
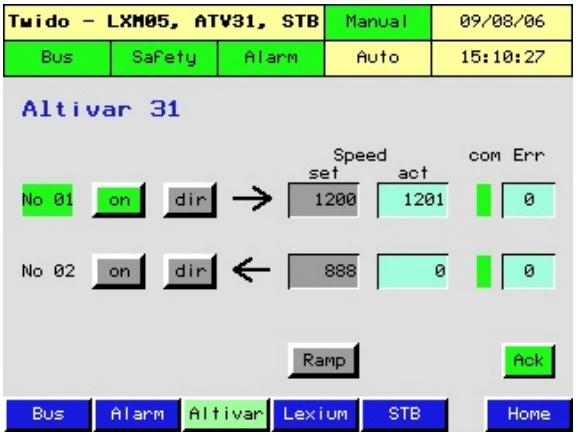
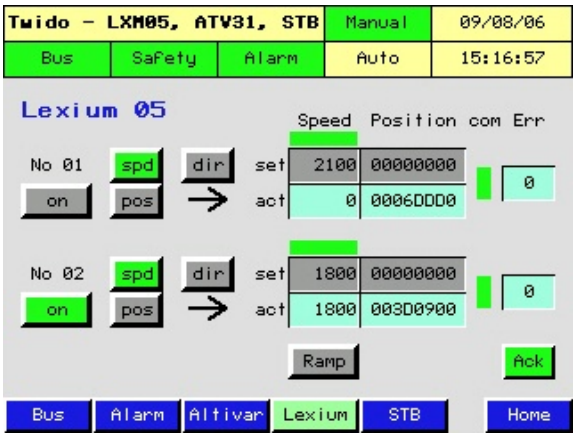
1	The status of the CANopen devices is available in the PLC as a value from 1 to 15. We want to show this value as text on a HMI screen. To do this select the object **Message Display** and position it on the HMI screen.	
2	First, in the **Variable Properties** dialog for the variable set the I/O setting to **BIN** and **16** bits. and in the attributes In the **Message Display Settings** input the **variable** name and set **States** to **16.** Now click on the **New Resource** icon (at the end of the **color Resource** field).	
3	In the **New Resource** dialog, input: Color name: **CANopenColor** Text Name: **CANopenStatus** Num.of states: **16** Check the box for **Message Display.** Data Type: **integer** and exit with **OK.**	

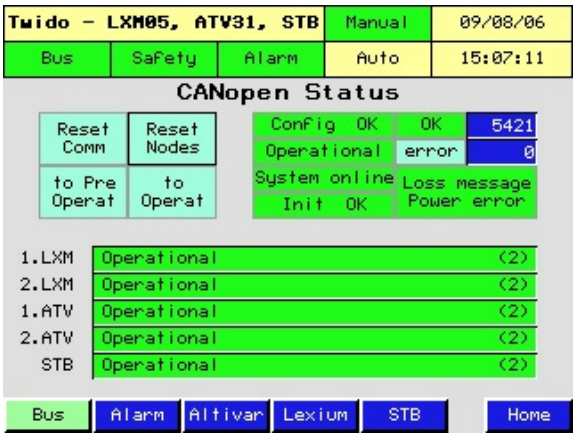
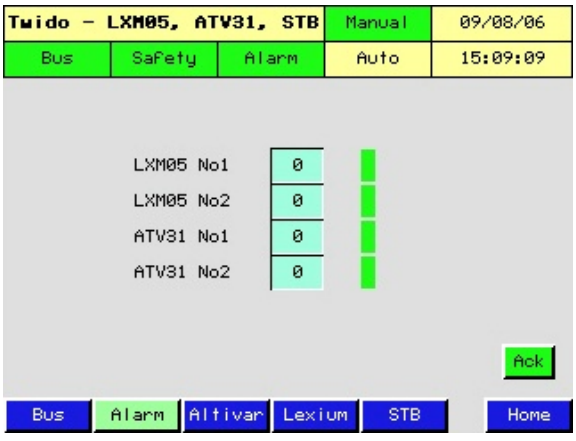
4	The **CANopenColor** defines different colours to be displayed according to the individual values of the variable. In the table that now appears you can define different colours for each of the 15 values and also define a text to be displayed with each value.	
		
5	The bus screen has a **message display** object for each device on the bus	
6	Once operational, the **message display**s show the actual condition of the device in the defined colours.	

Build and Download the Project

1	**Validate All** can be used to analyze the project. The Feedback Zone lists the results. The same applies to the **Build All** menu item.	 
2	Download the project to the Magelis (HMI): Select the project in the **navigator**. Use the menu item **Download All**, via right mouse click on the project name or the **Build** menu, to transfer the application to the connected HMI. The configured connection (serial or Ethernet) is used.	
3	**Defining the HMI device Ethernet IP-Address** If the project has not been already downloaded using a USB cable, the HMI has no valid IP address. This must be input to the HMI device in offline mode before the first download is attempted To do this: • Touch the HMI Screen on the top left hand corner to go to offline mode (if an application is already running, touch any 3 corners of the screen. The method used to go to offline mode can be defined in Vijeo Designer in the Platform Attributes menu) • Input the IP-Address. • Return to Online Mode.	

Application Summary

1	the example application has several screens that can be selected by the user. The starting screen shows the system structure. Manual mode is the default setting. The PLC is not configured for automatic mode. All drives can be controlled manually by changing to the screen for the device. All the following screens have the same control display at the top of the screen	
2	This image shows 2 altivar 31s. Each has a start/stop and direction button. Using SET the set point for the speed can be input The display shows the status, actual speed and any error codes for the drive.	
3	Here is the equivalent screen for 2 Lexium05 servo drives. Here there is a choice between speed(spd) and position(pos) modes.	

4	If a CANopen device fails, it is indicated in the header with the Bus indicator. If the user changes to the **Bus** screen (button bottom left) he can view the text message of the fault in the device. The status of the master module is also given. The bus can be reset using the button **Reset Nodes.** This resets all nodes. The communication is started with the **to Operat** button. all devices in pre-operational mode will go to operational mode. **Note:** **„Reset Node“ sets the parameters back to their factory values (e.g. Ramp with LXM05). These must be set up again with SDOs.**	
5	Drive errors are indicated by the Alarm indicator. Use the button **Alarm** (bottom left) to go to the screen for details of the alarm. Use the acknowledgement button (**ACK**) to acknowledge the alarm for ALL drives. . .	

PLC

Introduction

The PLC chapter describes the steps required for the initialisation and configuration and the source program required to fulfil the functions.

Pre-conditions

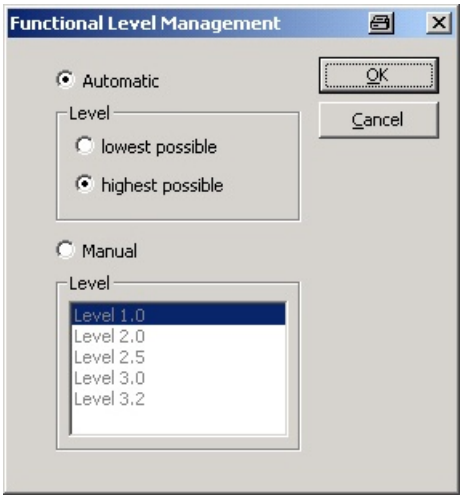
In order to proceed you require the following:

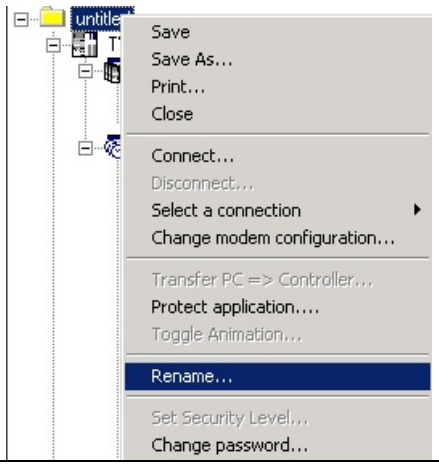
- Twidosoft installed on your PC
- The Twido PLC is switched on and running
- The PLC is connected to the PC with the programming cable (TSXPCX1031)

Proceed as follows to integrate the PLC:

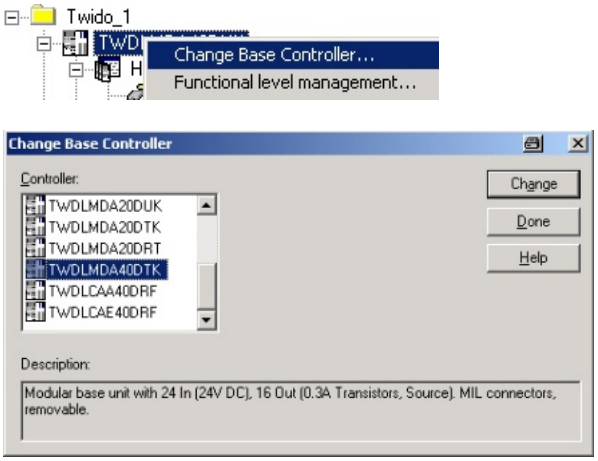
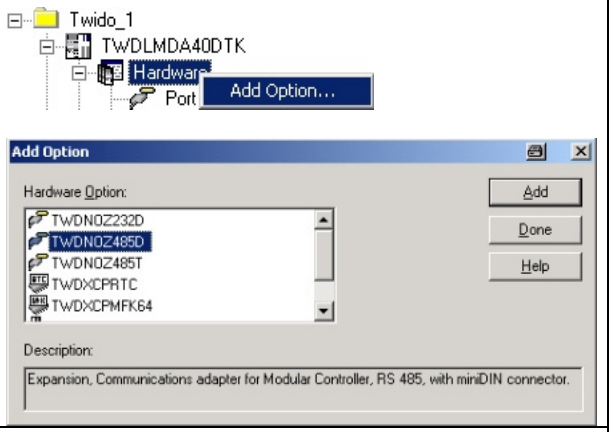
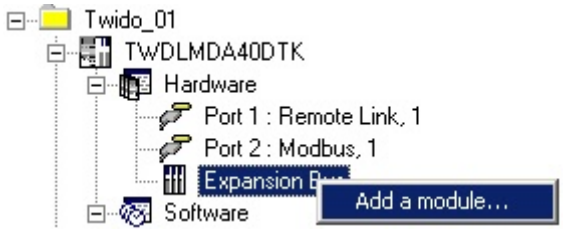
- Create a project
- Configure the hardware (central unit + modules)
- Configure Modbus communication
- Configure CANopen communication
- Create the Application Program
- Connect the PLC to the PC
- Transfer the user program to the PLC

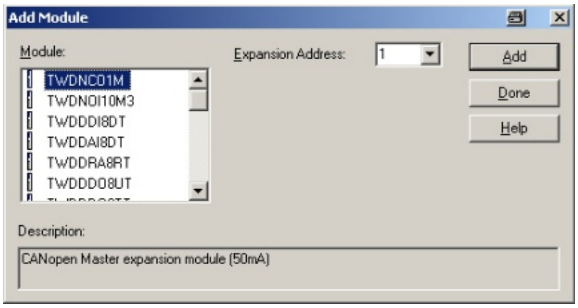
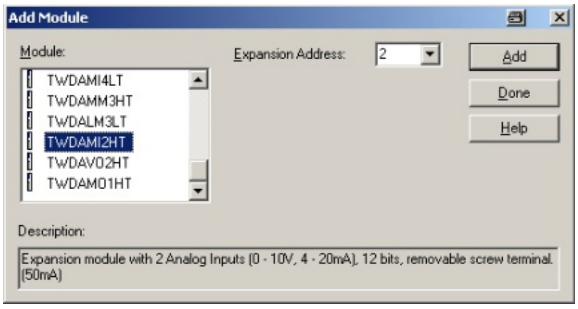
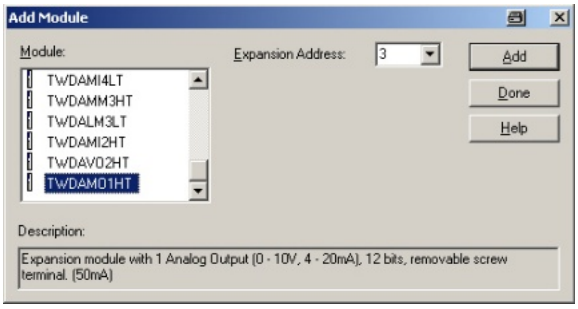
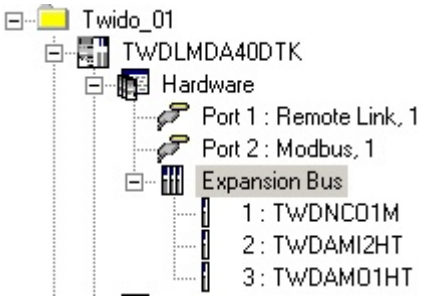
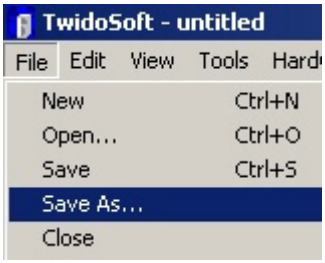
Create a project

1	After starting the software, the first thing you need to do is create a new project using **File->New**	
2	Select the functional level. You can leave the default value.	

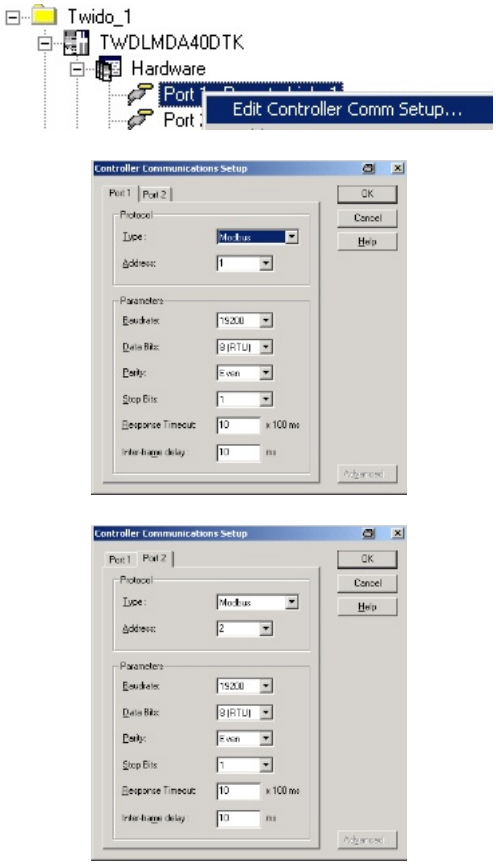
3	Give the application a name by right clicking on the default name in the project browser and selecting **Rename..**	

Configure the hardware

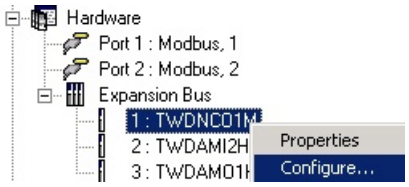
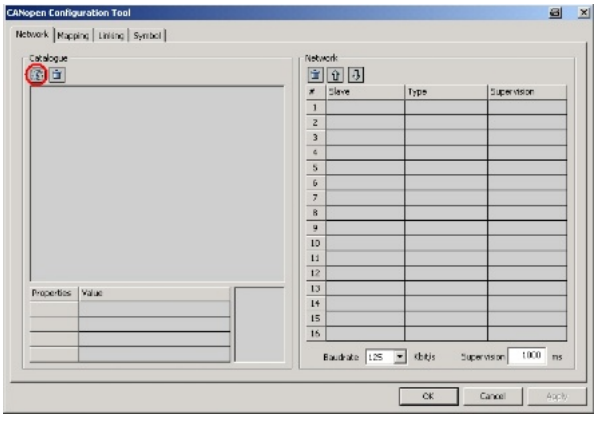
1	If the power base that is offered is not the one used you must change it by selecting: **change Base controller**... in the dialog, select the PLC power base you require and click on **Change.**	
2	Then add the additional RS485 interface by right mouse click on **Hardware** in the project browser and then **Add Option...**	
3	Similarly add the modules for the bus expansion	

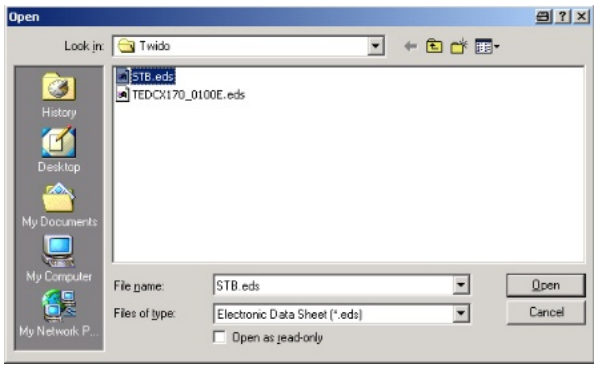
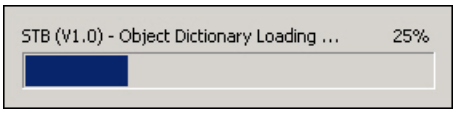
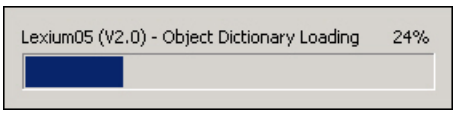
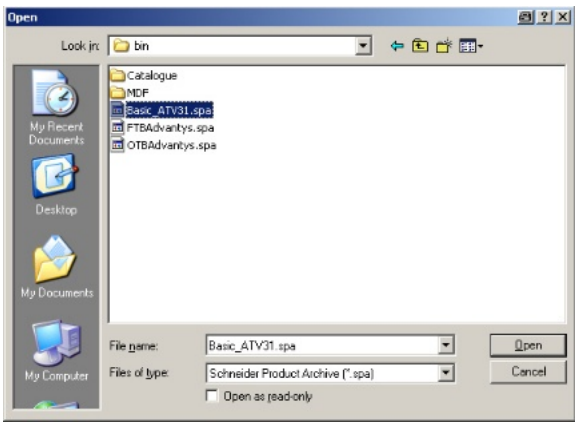
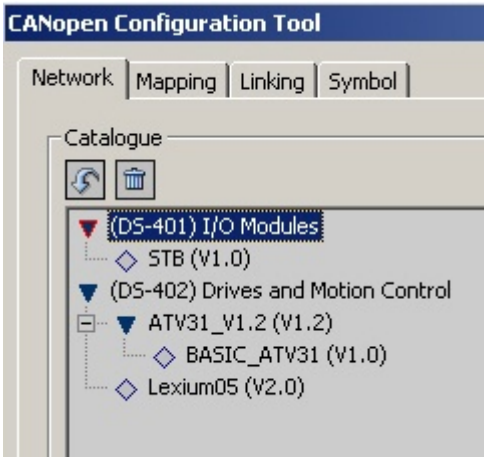
4	the CANopen Master **TWDNCOM1** and in this example, the analog I/O modules: **TWDAMI2HT** **TWDAMO1HT**.	  
5	The modules are now visible in the project browser.	
6	You can now save the project using: F**ile->Save as..** and later, at any time, using **File->Save**	

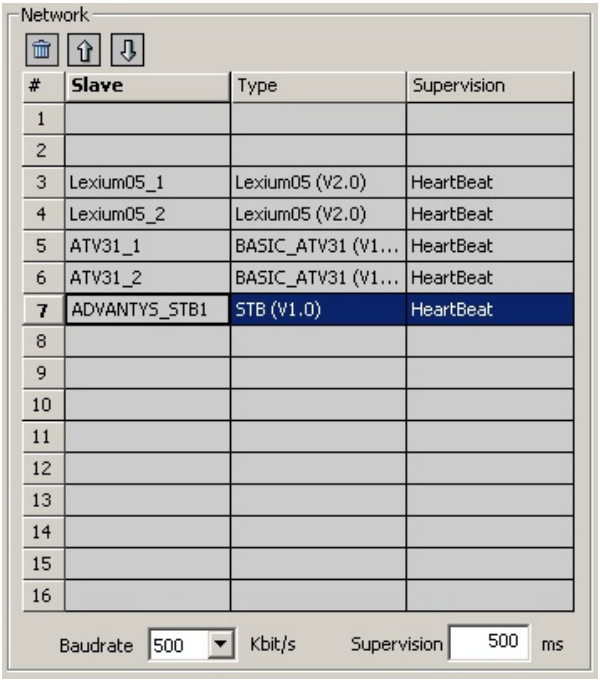
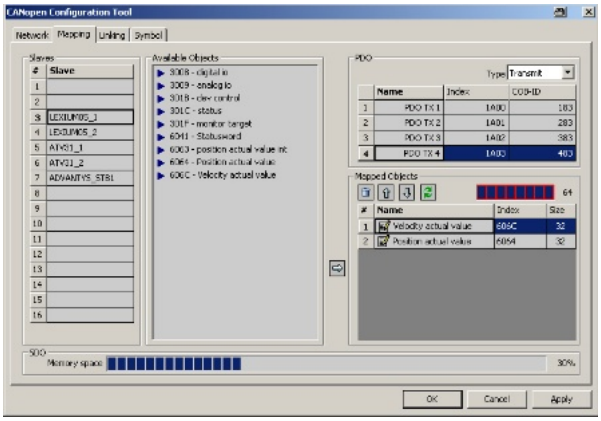
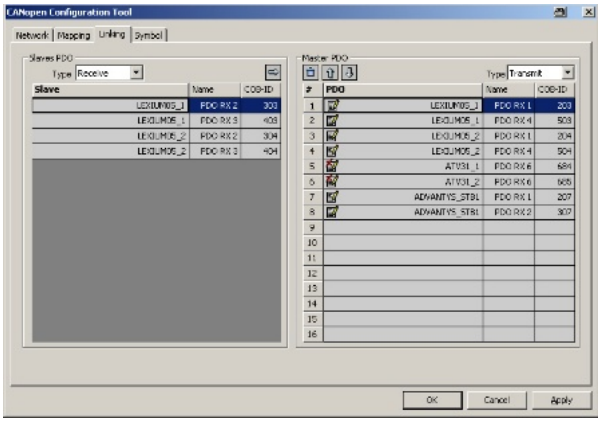
Configure Modbus Communication

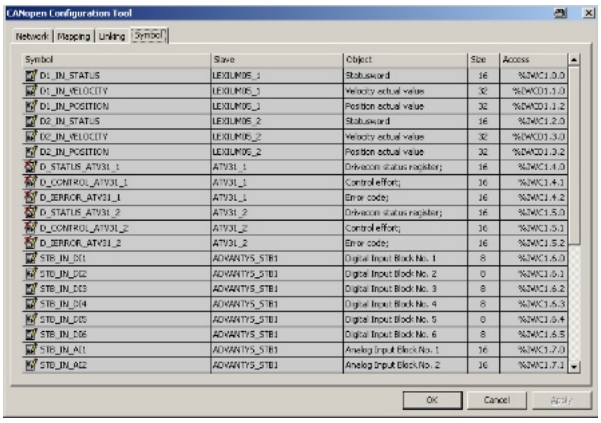
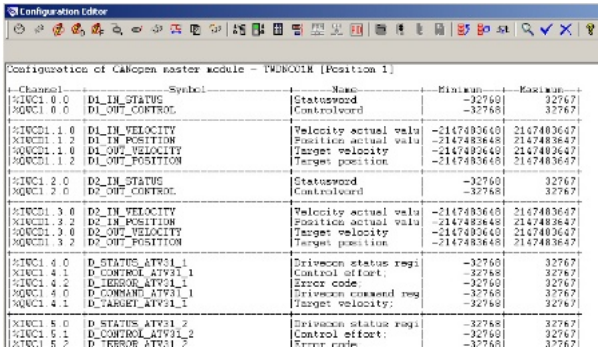
1	The appropriate interface parameters need to be entered. Communication type: **Modbus** Address: **1 for PC (Port 1)** **2 for HMI (Port 2)** Baud rate: **19200 baud** **Note**: The settings for Port 2 must be the same as those for the HMI.	

Configure CANopen communication

1	The configuration tool must be opened to configure the CANopen Master. In the project browser, right mouse click on the CANopen module TWDNCO1M and select: **Configure..**	
2	First the EDS files must be imported. Click on the **import/update** button in the window.	

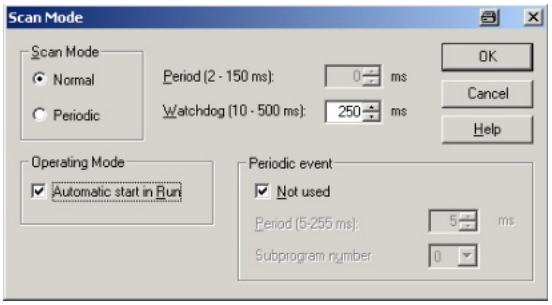
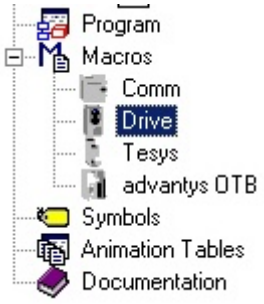
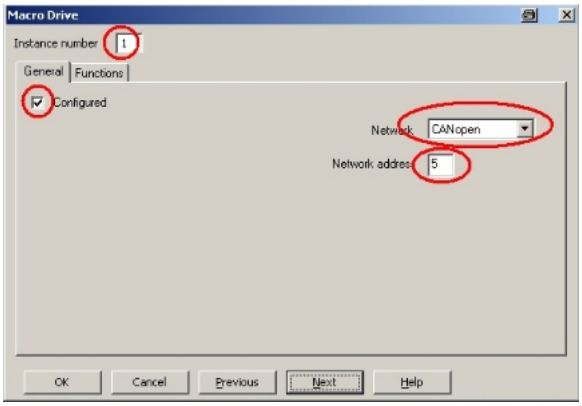
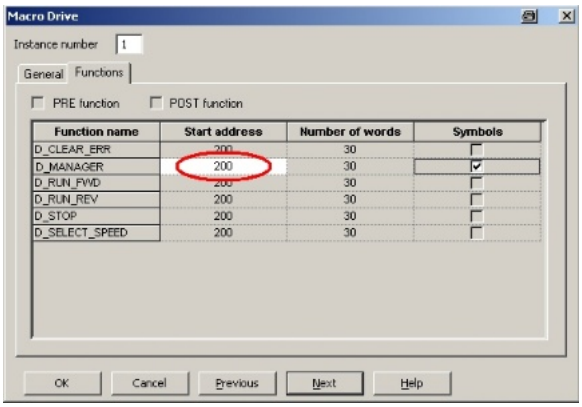
3	Next select, firstly the EDS file created by the Advantys software. Then select the EDS file that belongs to the lexium05 drive controller.	  
4	For the altivar31, change the file type to *.SPA and select the spa file for the altivar31. **Note:** the file can be found in the installation path: C:\Programme\Schneider Electric\TwidoSoft\bin	
5	All imported devices appear in the **Catalogue**. A double click on an entry creates a new entry in the **Network** on the right.	

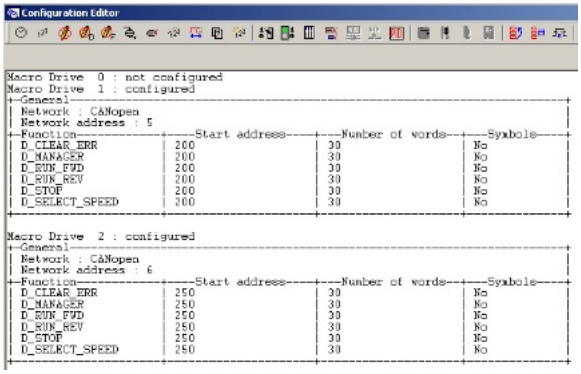
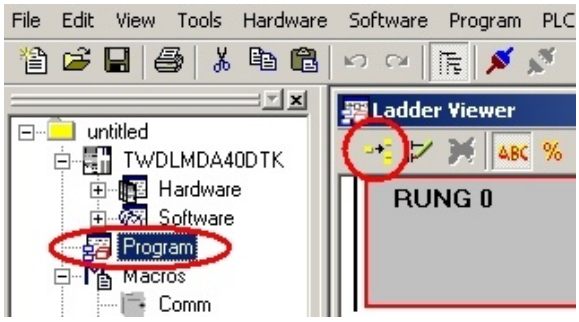
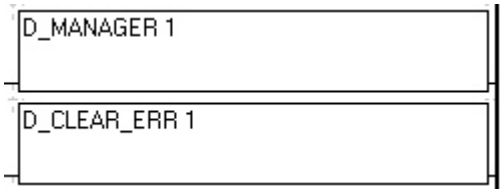
6	Slave Positions 1...16 correspond to the **CANopen-Address**. The slave position can be changed in the list using the arrow buttons. As the monitoring method, **Heartbeat** must be selected in the column **Supervision.** Set the **Baudrate** to **500 kbit/s** and the **Supervision** time to **500ms**. If you wish you can change the names of the slaves in the **Slave** column.	
7	On the **Mapping** tab you can view the pre-configured data. You must input the set and actual speed and position for the **PDO4** for the Lexium 05.	
8	The **Linking** tab shows which objects are linked to which device (PDO with Identifier). Selecting the individual PDOs moves them to the list on the right. The following PDOs are required (Receive/Send): **Lexium 05 TX1+4 RX1+4** **Altivar 31 TX6 RX6** **Advantys TX1+2 Rx1+2**	

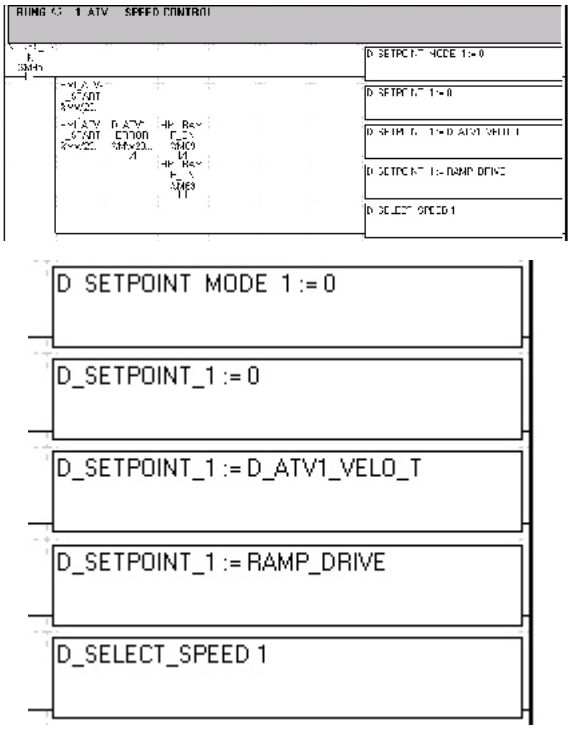
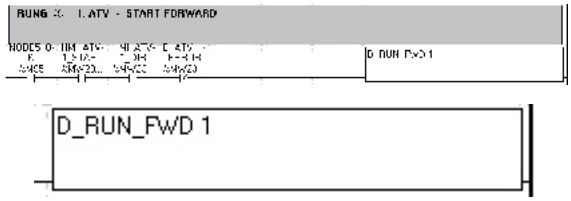
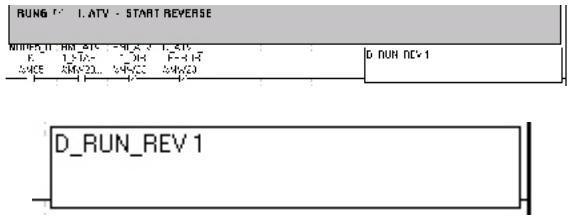
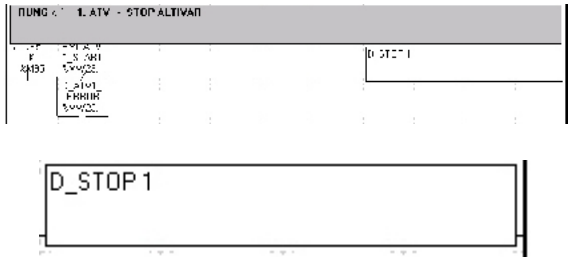
9	On the **Symbol** tab you can view the pre-defined symbol names from the configuration. Click on **OK** to accept the names and finish the CANopen configuration	
10	Twidosoft now displays a list of configured devices in bus address order.	

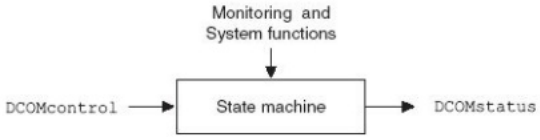
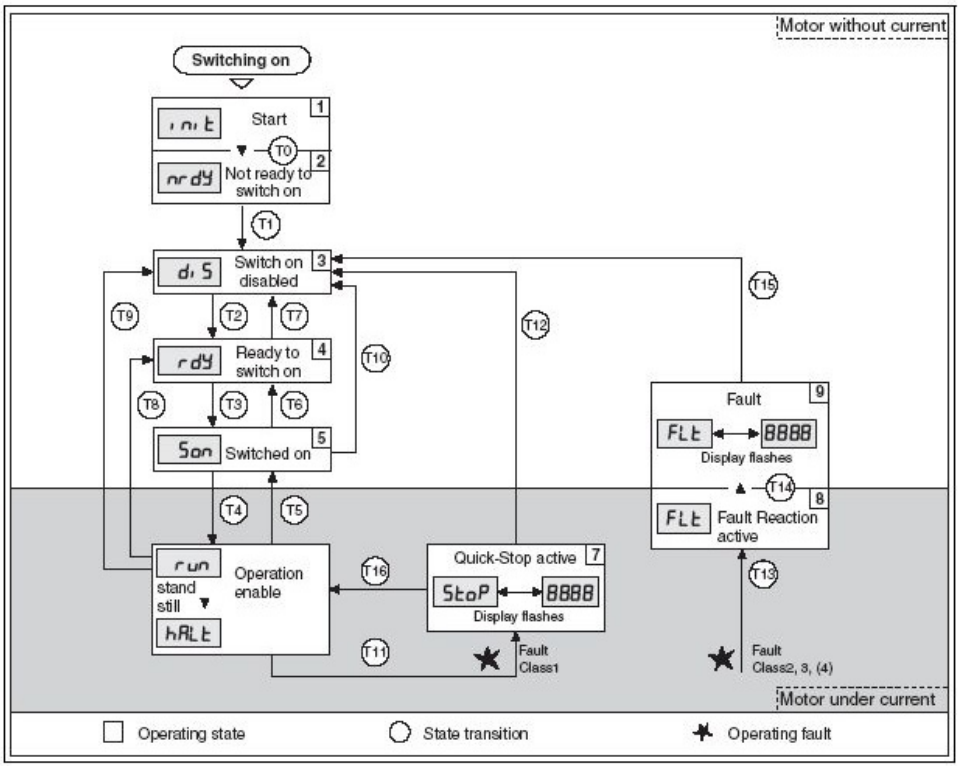
Continued on next page

Creating the application program

1	By selecting: **Program->Process Scan Mode** You can, amongst other things, select the autostart **operating mode.** Once you have set this you can go on to create an application program.	
2	To program the drives easier, Twidosoft has pre-defined macros for use with the Altivar31. A double click on : **Macros->Drive** opens up the configuration dialog	
3	In the **Macro Drive** dialog on the **General** tab, set the check box for **Configured** and input: **Network:** CANopen Then, for each altivar: 1. Altivar 31 **Instance number:** 1 **Network address:** 5 2. Altivar 31 **Instance number:** 2 **Network address:** 6	
4	Each drive requires 30 registers for each macro. On the **Functions** tab, input the **Start address** for each register group: **Start address** 200 for 1. ATV31 **Start address** 250 for 2. ATV31 Setting the check box in the **Symbols** column automatically generates the symbol names and inserts them into the symbol editor.	

5	When finished the configured macros can be viewed in the configuration editor.	
6	Now the code for controlling the drives must be produced. We do this in Ladder programming language. First create a new program network: Double mouse click on **Program** in the project browser to open the ladder editor with an empty section. Insert rungs into the section using the insert icon.	
7	Insert a block and a link for the first Altivar. Double click on the block and insert the name D_MANAGER 1 This block will invoke the first instance of the macro. Insert another block and enter the command D_CLEAR_ERR 1. The conditions under which a macro is invoked (here a reset) depend on the application and are will not be explored here. Note: The connection to the individual drives is made using the instance number, here 1 1 = 1. Instance = 1. ATV31 2 = 2. Instance = 2. ATV31	 

8	This network contains several blocks to set the set point of drive 1.	
9	The command RUN_FWD 1 puts drive 1 into forward motion	
10	The command D_RUN_REV 1 puts drive 1 into reverse. Note: A drive runs in reverse when started forwards but with the set point negative.	
11	Once started drive 1 can be stopped using the command D_STOP 1 n. Use the same commands for instance 2 to control drive 2 (D_STOP 2 etc.)	

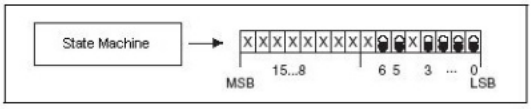
12	The **Lexium05** shows the relationships between the operating states and state transitions in the state machine. The operating states are influenced by the user with the control word **(`controlword`)** and monitored with the status word (**`statusword`**).	
13	The Lexium05 displays the operating states numbered **1** to **9** in rectangles and the transitions numbered 0 to 16 in circles. 	
14	When it is switched on the Lexium05 is in state 4 (rdy) and when the drive is running it is in state 6 (run). Description of operating states:	

Display	Status	State description
init	**1** Start	Controller supply voltage, electronics is initialised
nrdy	**2** Not ready to switch on	The power amplifier is not ready to switch on
di S	**3** Switch on disabled	Switching on the power amplifier is disabled
rdy	**4** Ready to switch on	The power amplifier is ready to switch on
Son	**5** Switched on	Motor not under current Power amplifier ready No operating mode active
run hALt	**6** Operation enable	RUN: device running in the selected operating mode HALT: The motor is stopped with active power amplifier
StoP	**7** Quick Stop active	"Quick Stop" is executed
FLt	**8** Fault Reaction active	Error detected, error response is enabled
FLt	**9** Fault	device is in error condition

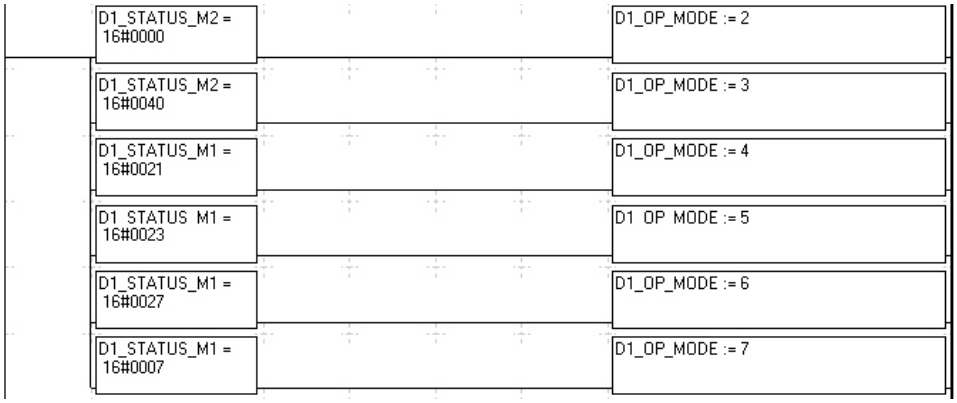
15

When standardized operating modes are in use, the operating states are monitored via bits 0 to 3, 5 and 6 and the status word.

The status word is read in via the CANopen bus and the operating state is written in %MW120 (%MW170 for the second Lexium05).



Status	Bit 6 Switch On disable	Bit 5 Quick Stop	Bit 3 Fault	Bit 2 Operation ENABLE	Bit 1 Switch On	Bit 0 Ready to Switch On
2: Not ready to switch on	0	X	0	0	0	0
3: Switch on disabled	1	X	0	0	0	0
4: Ready to switch on	0	1	0	0	0	1
5: Switched on	0	1	0	0	1	1
6: Operation enable	0	1	0	1	1	1
7: Quick Stop active	0	0	0	1	1	1
9: Fault	0	X	1	1	1	1

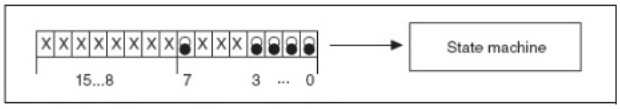


16 State transitions are triggered by a command or in reaction to a monitoring signal. A command is given to the Lexium05 via the **`controlword.`**

State transitions 0, 1 and 14 occur automatically in the device and are not command-activated. The following table shows state transitions that can be triggered by commands.

Transition	Operating status	Condition / result [1]	Response
T0	1 -> 2	• Motor speed below switch-on limit • Device electronics successfully initialised	Check motor encoder
T1	2 -> 3	• First commissioning is completed	-
T2	3 -> 4	• Motor sensor successfully checked, DC bus voltage active, $\overline{\text{PWRR_A}}$ and $\overline{\text{PWRR_B}}$ = +24V, actual speed: <1000 rpm fieldbus command: Shutdown [2]	-
T3	4 -> 5	• Fieldbus command Switch On • Input signal ENABLE0 -> 1	
T4	5 -> 6	• Fieldbus command Enable Operation	Switch on power amplifier. Motor phases, earthing, user parameters are checked Release brake
T5	6 -> 5	• Fieldbus command Disable Operation • Input signal ENABLE0 -> 1	Interrupt travel command with "Halt" Apply brake Switch off power amplifier
T6	5 -> 4	• Fieldbus command Shutdown	
T7	4 -> 3	• DC BUS low voltage • $\overline{\text{PWRR_A}}$ and $\overline{\text{PWRR_B}}$ = 0V • Actual speed: >1000 rpm (e.g. by auxiliary drive) • Fieldbus command Disable Voltage	-
T8	6 -> 4	• Fieldbus command Shutdown	Switch off power amplifier immediately
T9	6 -> 3	• Fieldbus command Disable Voltage	Switch off power amplifier immediately
T10	5 -> 3	• Fieldbus command Disable Voltage	
T11	6 -> 7	• Class 1 error • Fieldbus command Quick Stop	Interrupt travel command with "Quick Stop"
T12	7 -> 3	• Fieldbus command Disable Voltage	Switch off power amplifier immediately, even if "Quick Stop" still active
T13	x -> 8	• Class 2, 3 or 4 error	Error response is carried out, see "error response"
T14	8 -> 9	• Error response completed • Class 3 or 4 error	
T15	9 -> 3	• Fieldbus command Fault Reset [3] • Input signal FAULT_RESET0 -> 1 [3]	Error is reset
T16	7 -> 6	• Fieldbus command Fault Reset [3] • Fieldbus command Enable Operation [4] • Input signal FAULT_RESET0 -> 1 [3]	Local control mode specified operating mode is automatically continued

17	The operating states are set via the control word. Bits 0 to 3 and bit 7 are relevant to state transitions. The bit states in the fields marked “X” are not relevant to the state change concerned.



Fieldbus command	state transitions	Change of state to	Bit 7, Reset Fault	Bit 3, Enable operation	Bit 2, Quick-Stop	Bit 1, Disable Voltage	Bit 0, Switch On
Shutdown	T2, T6, T8	**4**: Ready to switch on	X	X	1	1	0
Switch On	T3	**5**: Switched on	X	X	1	1	1
Disable Voltage	T7, T9, T10, T12	**3**: Switch on disabled	X	X	X	0	X
Quick Stop	T7, T10T11	**3**: Switch on disabled**7**: Quick Stop active	X	X	0	1	X
Disable Operation	T5	**5**: Switched on	X	0	1	1	1
Enable operation	T4, T16	**6**: Operation enable	X	1	1	1	1
Fault Reset	T15	**3**: Switch on disabled	0 -> 1	X	X	X	X

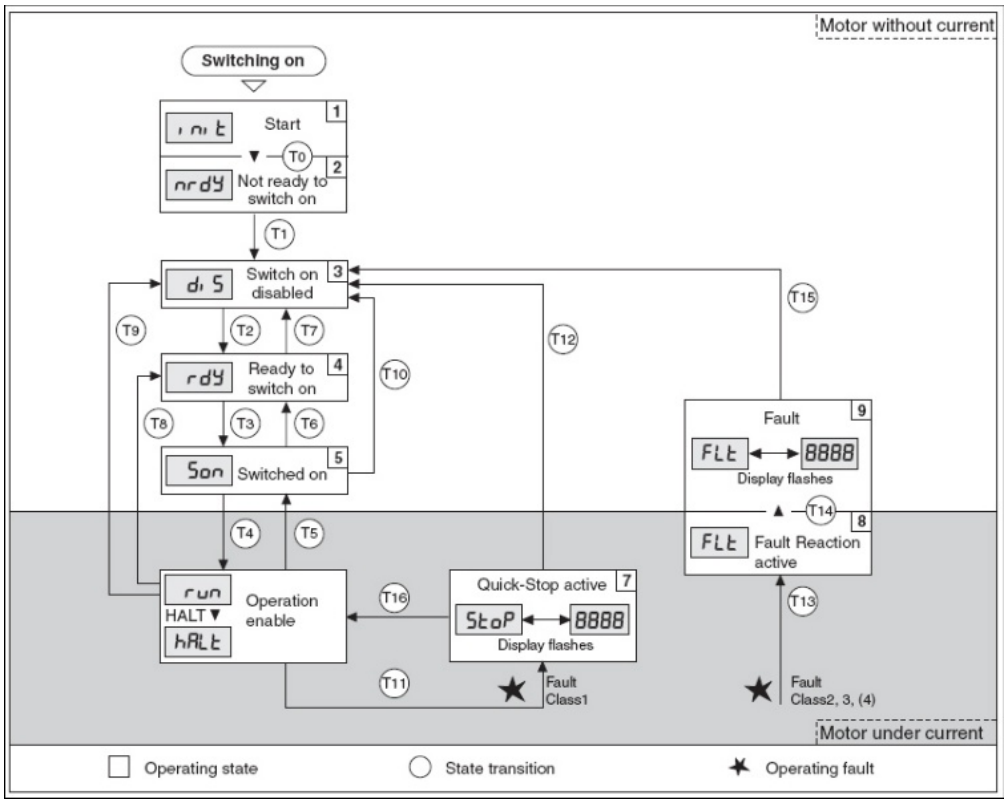
Continued on next page

Creating the application program
contd.

8 After power restoration the Lexium05s are designed to return automatically to operating state 4 **(rdy)** “Ready to switch on”.

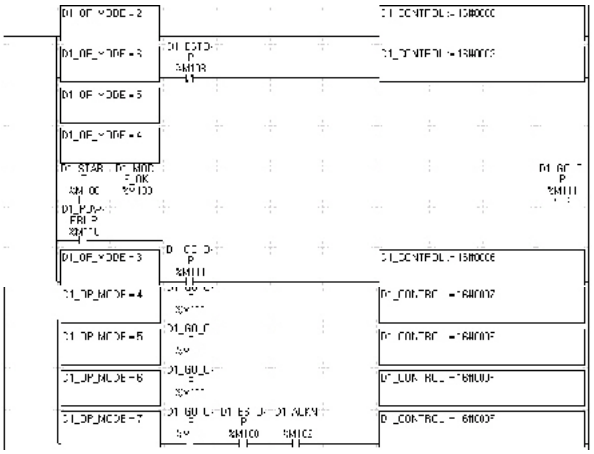
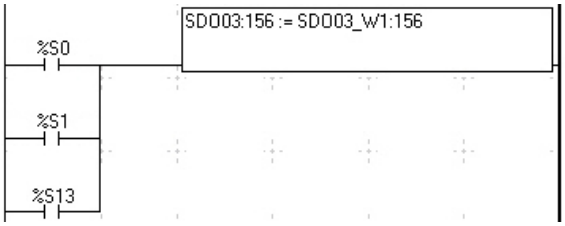
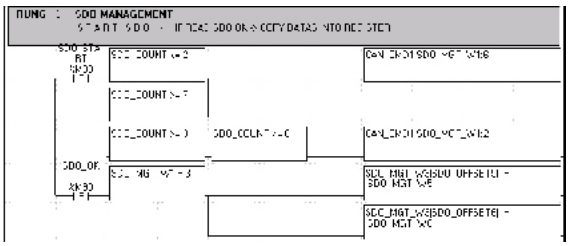
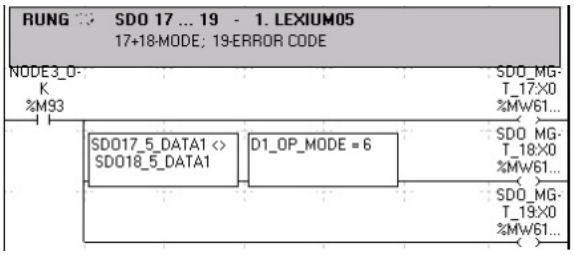
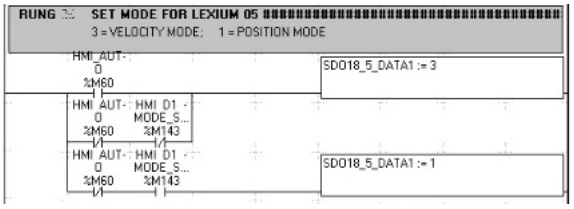
This can also be tracked on the Lexium05 display.

Display	State description
init	Controller supply voltage, electronics is initialised
nrdy	The power amplifier is not ready to switch on
di S	Switching on the power amplifier is disabled
rdy	The power amplifier is ready to switch on
Son	Motor not under current Power amplifier ready No operating mode active
run hALt	RUN: device running in the selected operating mode HALT: The motor is stopped with active power amplifier
StoP	"Quick Stop" is executed
FLt	Error detected, error response is enabled
FLt	device is in error condition



Continued on next page

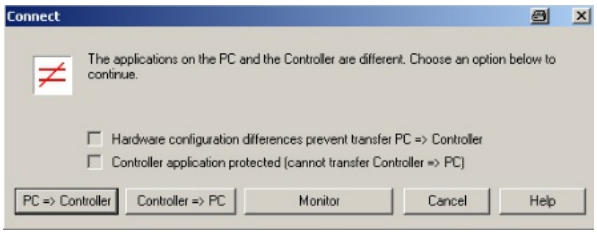
Creating the application program contd.

19	To change the drive to operating state 6 (**run**) "Operation enable", activate the PLC logic by pressing the "Power up" button on the Magelis HMI. This causes the state machine to run in sequence.	
20	As well as cyclic Process Data Objects (PDOs), CANopen also has Service Data Objects (SDOs) The command CAN_CMD is used to send SDOs. This application uses 28 SDOs. The SDO data is moved from the constant memory (%KW) to memory words (%MW) during system initialisation.	 
21	In this application the "speed profile" and "point to point" modes are used (see also PDO 4). Since the Lexium05 may be in a different operating mode, e.g. after power restoration, it must be ensured that the proper profile is activated. The end stage must be switched on (operating state 6, "Operation enabled") in order to change the mode. If it is, the instruction CAN_CMD can be used to write "**03**" for speed profile, or "**01**" for point to point", to the mode register 6060:0_{hex} of the Lexium.	 

Creating the application program contd.

22	The fault register 603F:0_{hex} of the Lexium05 is read out at regular intervals. The **CAN_CMD** instruction is also used for this purpose. After the Advantys STB is switched on, the data exchange from analog inputs as per CANopen guideline is deactivated. It has to be enabled via the Advantys register 6423:$0_{hex.}$ The **CAN_CMD** instruction is also used for this purpose.
23	Only one SDO can be active at any one time. The state can be monitored with %SW81.

Connect the PLC to the PC and download the program

1	Define the connection to the PLC. In this example we use **COM1**.	
2	Connect the PLC to the PC For this to work, the appropriate programming cable must be connected to the PLC.	
3	Transfer the user program to the PLC. Connecting to the PLC triggers an automatic check that the application program is the same as the one in the PLC. If it is not the case, a download or upload is proffered. If download is selected, you must click **OK** to confirm the run-> stop change and overwriting of the existing program.	
4	After the download the PLC is in **STOP** mode. You must click the START button to start the program. Once started the status changes to **RUN** mode.	

Devices

Introduction

This chapter describes the steps required to initialise and configure the devices to attain the described system function.

General

The following devices are described:

- Advantys STB I/O Platform

  You require the Advantys configuration software to download the configuration to the Advantys and also to generate the EDS file for the PLC.

- Altivar31 and Lexium05

  You can configure the Altivars and Lexiums using the front panel on the device itself, however, using the software tool Powersuite has certain advantages:

  - Save data to your hard drive and duplicate it
  - Print documentation for your configuration
  - Test and optimise the parameters online.

I/O platform

Introduction

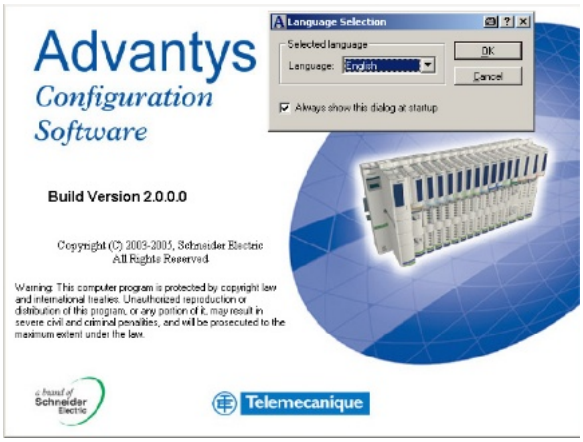
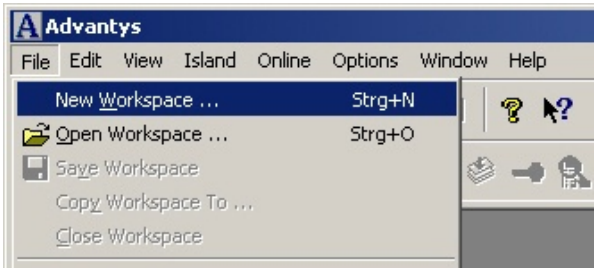
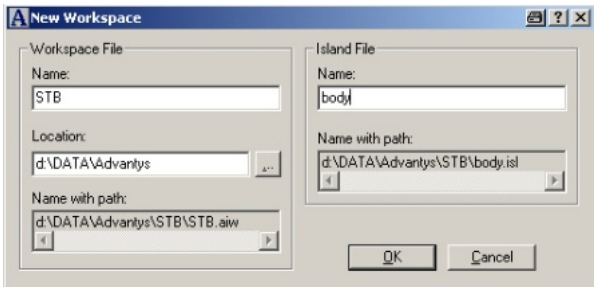
This section describes how the Advantys STB I/O platform is configured. The Advantys configuration software is used for this purpose.

We suggest that you proceed as follows:

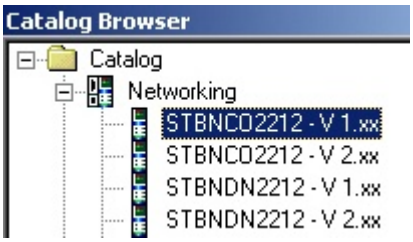
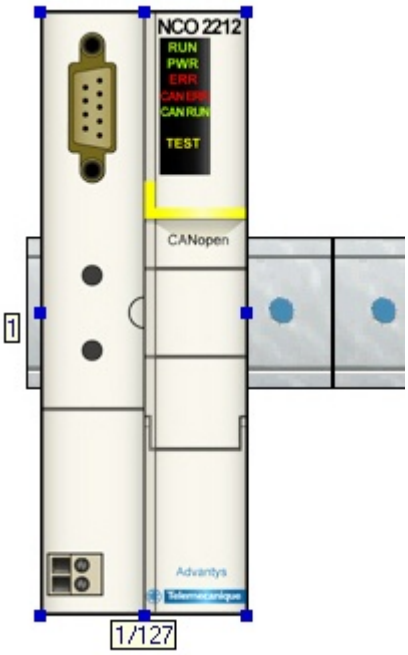
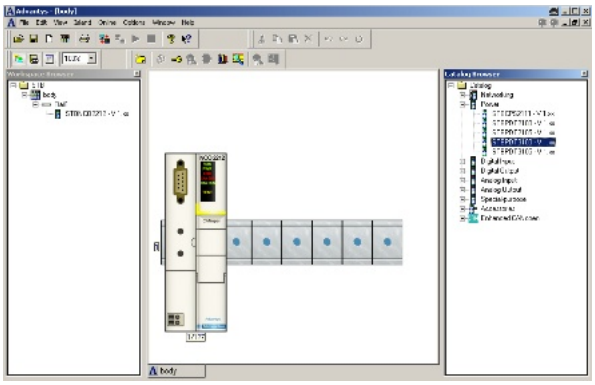
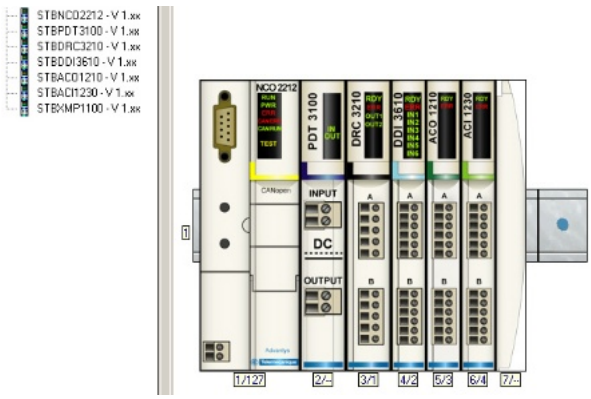
- Create a new project (workspace)
- Configure the hardware (network interface, power supply and I/O modules)
- Configure CANopen communication
- I/O assignment
- Download the Configuration
- Create an EDS file

Create a new project (Workspace)

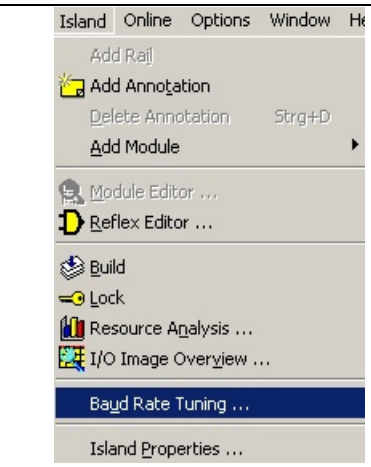
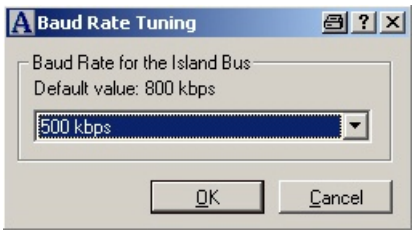
1	After installing the Advantys software you are given the choice of choosing between Advantys STB or Advantys FTB, FTM and OTB. Select **STB**.	

2	Choose the language in which you wish to work.	
3	After starting the Advantys configuration software, you must create a **New Workspace**.	
4	To do this, specify the path, the workspace name and the name of the first island.	

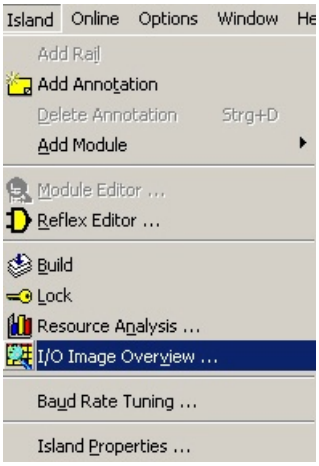
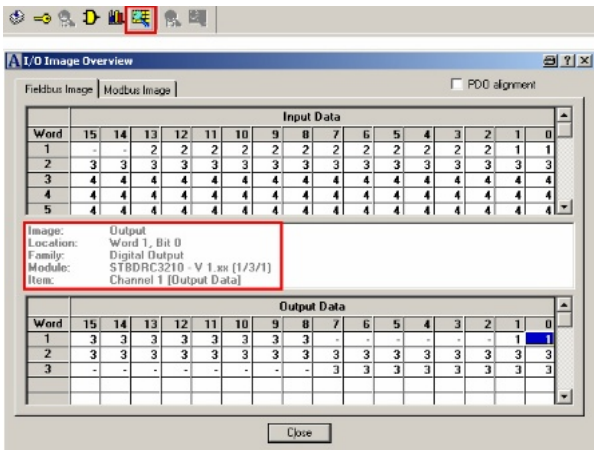
Configuring the hardware

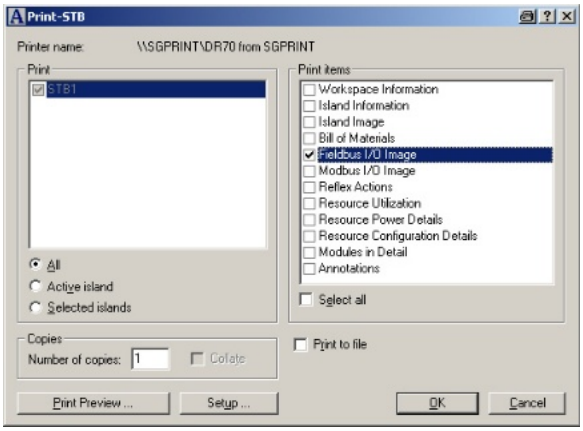
1	Select the modules for the island from the **Catalog Browser.** Select the network interface for CANopen: • **STB NCO 2212 –V1.xx**	 
2	Next, add the other stations: • **STB PDT 3100** • **STB DRC 3210** • **STB DDI3610** • **STB ACO 1210** • **STB ACI 1230** Remember the bus connection: • **STB XMP 1100**	
3	The display should now look like this.	

Configuring CANopen Communication

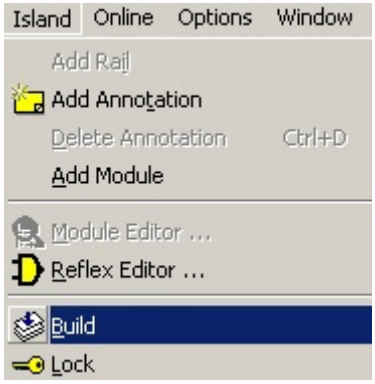
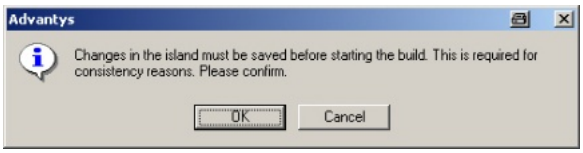
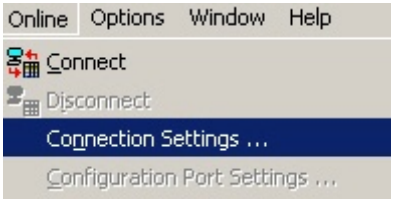
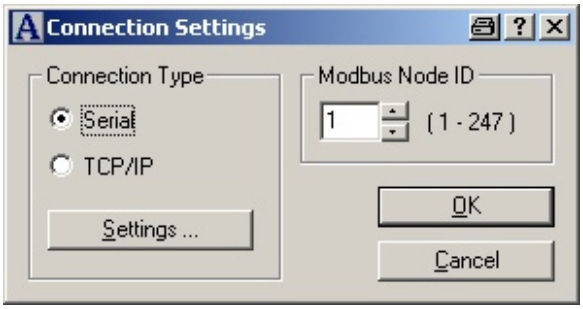
1	The internal baud rate can be set via the menu bar. The rate used is 500 kbps. Set the parameter for the transfer rate between NIM and PLC with the two rotary switches on the front of the NIM.	 

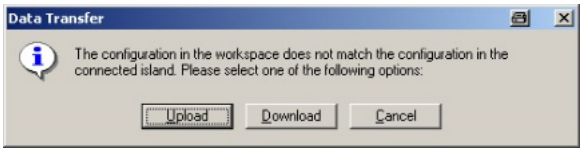
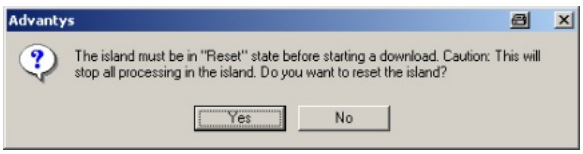
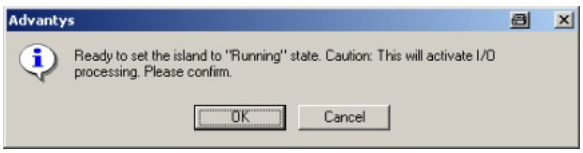
I/O Assignment

1	You can use the **I/O Image Overview...** menu item or the icon to call the function for assigning I/Os to memory areas.	
2	In the **Fieldbus Image** tab the information concerning the selected data is displayed in the description field.	

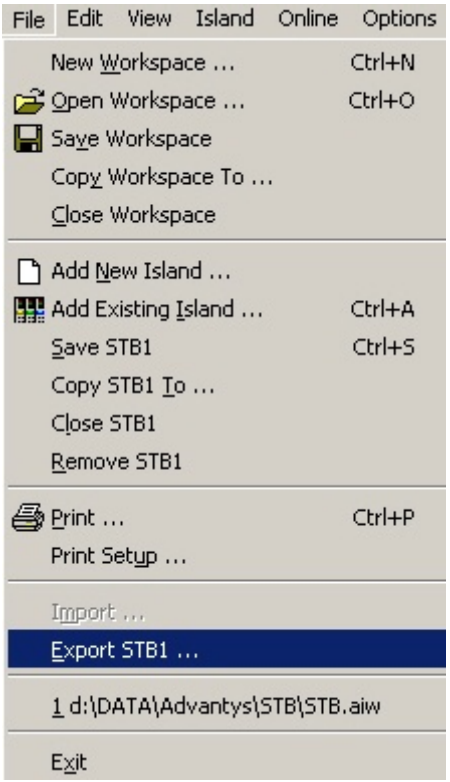
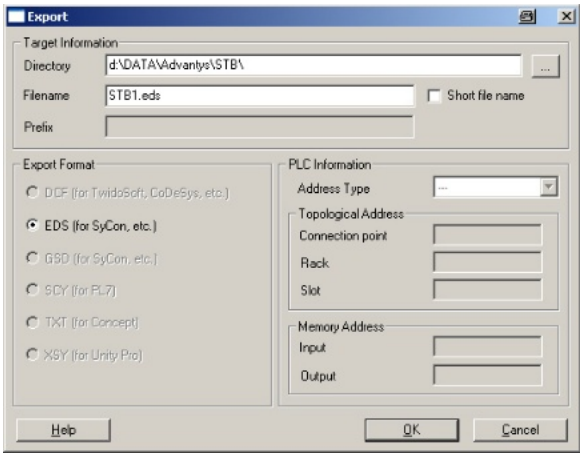
3	Alternatively, the project can also be printed out. The printout will contain the same information as the screen version.	

Download the Configuration

1	Before downloading the project it must be built For this select: **Island->Build**	
2	If you have not yet saved the project you will be asked to do so. Enter **OK**.	
3	You can check the status of the build in the protocol window. Check that the build was successful.	Island file has been saved. Build of island data in progress ... Build completed successfully.
4	To download the configuration first set up the connection via **Online->Connection Settings** select: **Connection Type**: serial and set the **Modbus Node ID** to 1.	 

5	Connect to the device via **Online->connect.** The configuration versions in the NIM and the PC will be compared. If there is a difference you will receive a warning. Continue with **Download**.	 
6	Acknowledge the following warnings with **YES** and **OK** You will be shown a progress bar once the download starts.	 
7	Once the download is finished you should see the island status as **healthy**.	**Island is healthy.**
8	Now you can disconnect from the device.	

Create an EDS File

1	To create an EDS file, select: **File->Export STB1...**	
2	In the **Export** dialog, enter the **directory** and the **filename** Exit with **OK.**	

Altivar 31

Introduction

The ATV31 device chapter describes how to initialise and parameterise the Altivar ATV31 devices in order to fulfil the system functionality described above.

PowerSuite software is used to initialise and parameterise the devices.

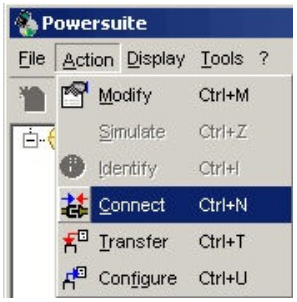
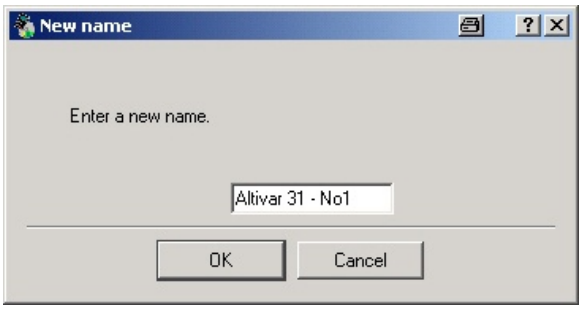
General

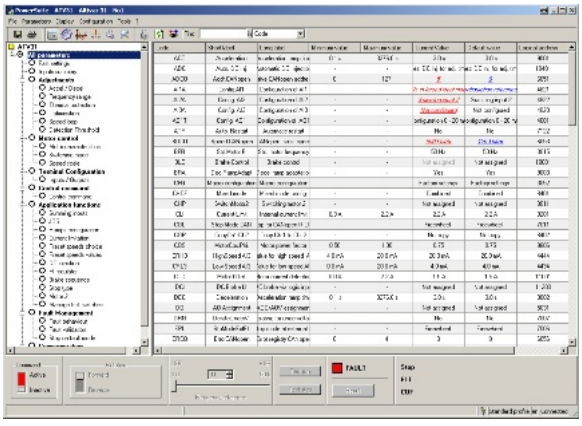
The ATV31 parameters can also be entered or modified via the front panel. The advantages of using PowerSuite are that you can:

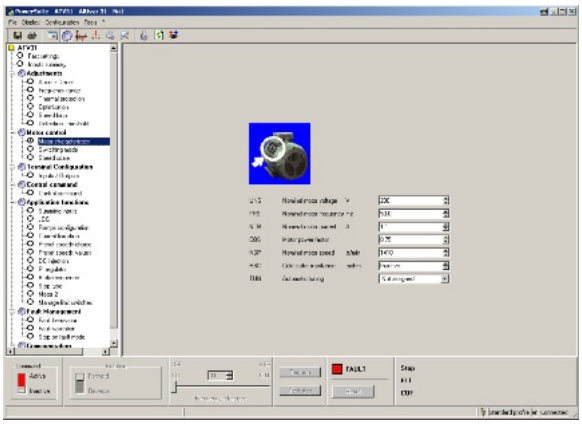
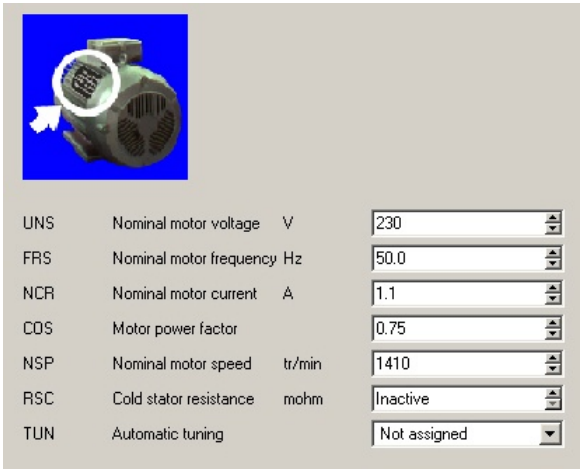
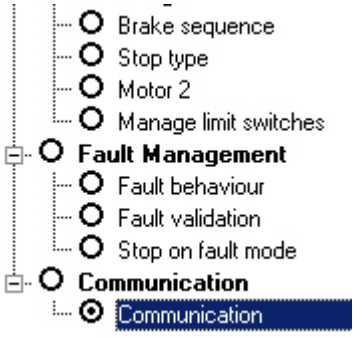
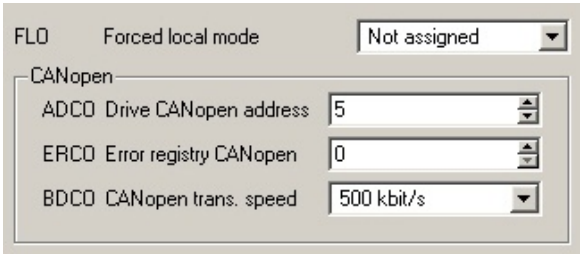
- save the data on your PC and copy it as you wish
- print out the documentation *and*
- be assisted in optimising the parameters online.

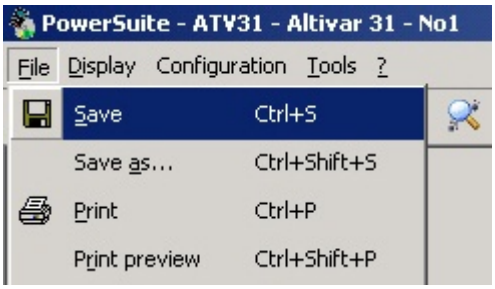
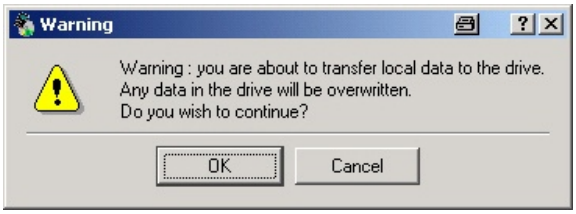
PowerSuite with ATV31

The Parameters can be configured with the Powersuite configuration software.
Here, the configuration was done using Powersuite V2.3

1	After starting Powersuite, select the entry **Example-folder**	
2	Via the menu **Action→Connect** connect to the device. Note: make sure the device is connected with the proper cable.	
3	It will be confirmed that you are trying to connect to a new device. Click on **Create** to continue.	
4	In the **New name** dialog give the configuration for the device a name.	

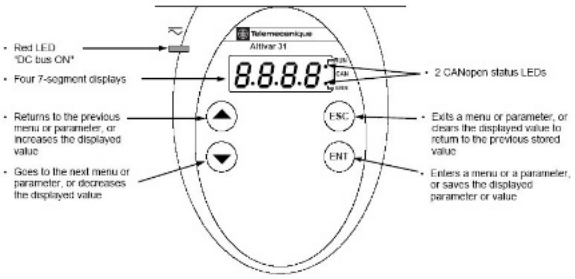
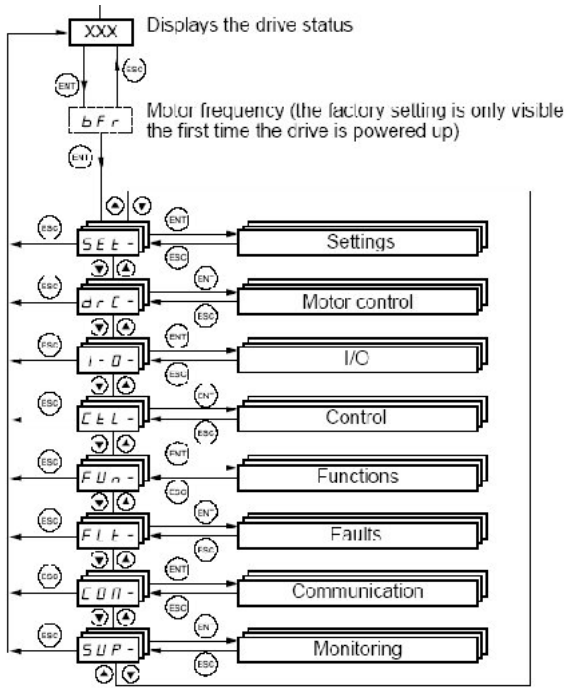
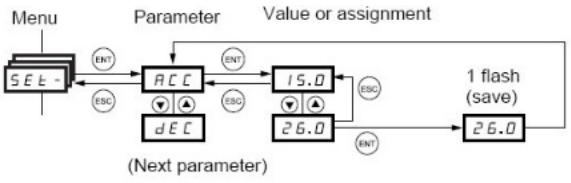
5	You will see a progress bar as the data is read from the Altivar 31 device.	
6	When the transfer is complete, the device data is displayed.	 **Altivar 31 - No1** **Characteristics** Reference: ATV31H018M2 Power: 0.18 kW / 0.25 HP Voltage: 200/240 V single phase Zone: Europe Hardware type: Product on heatsink Nominal current: 1,5 A Max. transient current: 2,3 A **Structure** Card / Reference / Serial number / Version / Vendor name Device: ATV31H018M2 / XX X5 52 025 045 / V1.7IE15 / TELEMECANIQUE Control Board: / XX Xf ff ff ff / V1.7IE15 / TELEMECANIQUE HMI Board: VX4A311/312(A) / XX Xf ff ff ff / V1.1IE02 / TELEMECANIQUE
7	You can view the parameter in list form or ...	

8	... in an input mask which can be viewed via: **Adjustments->** **Motor control ->** **Motor Characteristics**	
9	Enter the Motor data of the type of motors used.	
10	In the entry **Communication** enter the CANopen Address (for this application **5** & **6**) and select : Baudrate: 500 kbit/s.	 

11	**Save** the configuration.	
12	To transfer the new parameters to an individual device, right mouse click on the entry for the device and select: **download** in the pop-up menu.	
13	Confirm the warning message with **OK**.	

Manual Input using the Altivar Front Panel

You can configure the ATV 31 by inputting the parameters using the front panel display and buttons on each Altivar, as follows:

1	The CANopen-Address and Baudrate can be input using the buttons on the front panel of the Altivar.	
2	Using the buttons on the front panel, select the sub-menu **Communication** auszuwählen.	
3	In the **Communication** sub-menu input the CANopen address in the parameter **AdC0.** In the example application the adresses for the four controlers are 1 to 4.	
4	Also in the **Communication** sub-menu, in the parameter **BdC0,** set the baudrate to **500.0** (kBits).	
5	Alternatively you can use the PowerSuite software to configure the CANopen addresses and baudrates.	
6	**To activate the downloaded bus parameters (address and baudrate) you must now switch off the controller (display goes off). On switching back on, the new parameters are ready.**	

Lexium05

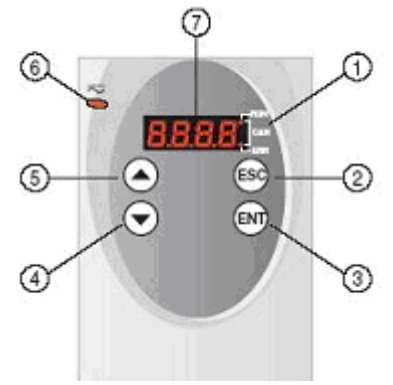
Introduction

This section describes the basic settings that have to be made on the Lexium05 drive controls.

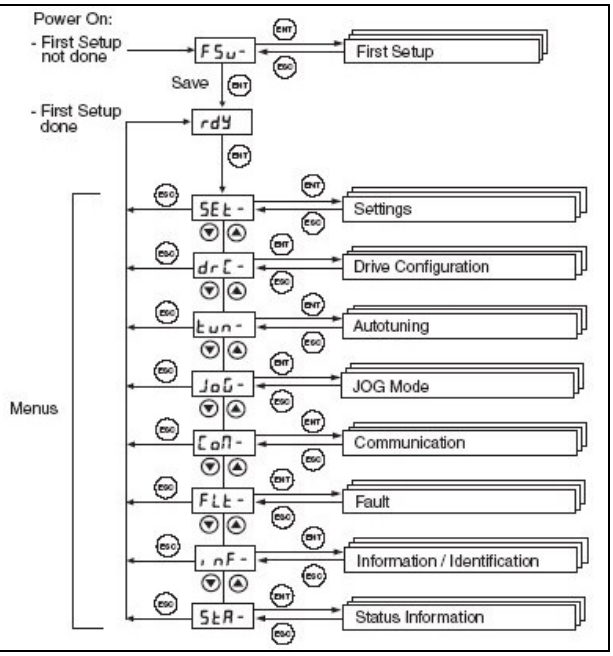
In particular, these include the communication parameters such as:

- Field bus type — CANopen
- Address — 3 or 4 in this instance
- Transfer speed — 500 kbaud

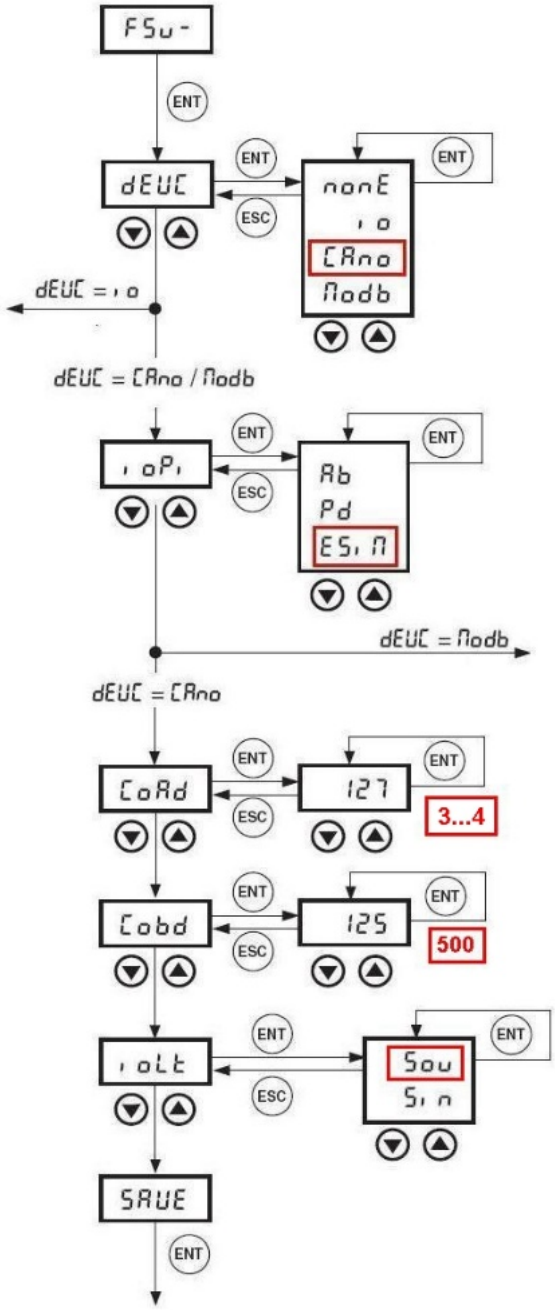
Basic settings

1	After wiring is complete the drive control parameters must be set. Parameters can be edited via the integrated operating panel (HMI).	 1 LEDs for CANopen 2 ESC - Exit from a menu or a parameter - Return to the last saved value 3 ENT - Call up a menu or a parameter - Save the displayed value 4 Down arrow - Change to the next menu or parameter - Decrease the displayed value 5 Up arrow - Change to the previous menu or parameter - Increase the displayed value 6 Red LED lit (DC bus live) 7 7-segment 4-character display

Continued on next page

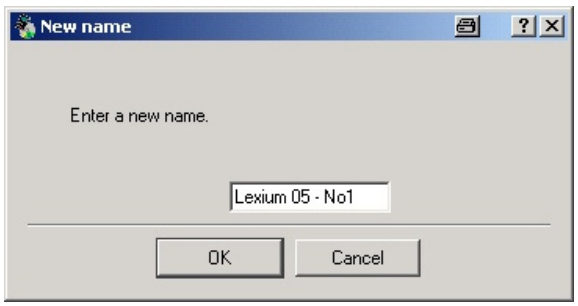
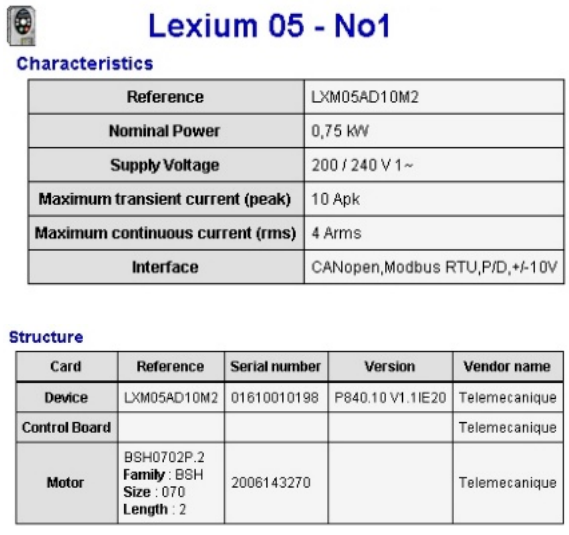
2	The HMI offers a menu system. The image here shows the top layer of the menu hierarchy In order to use the PowerSuite Software check the Modbus parameters. **CoM** is **MbAd** = **1** and **Mbbd** = **19.2**	

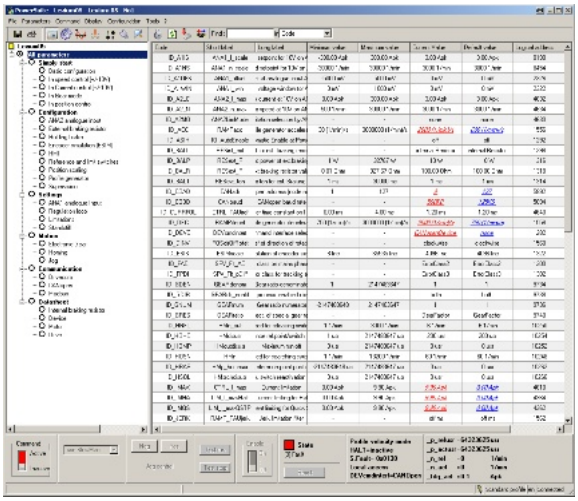
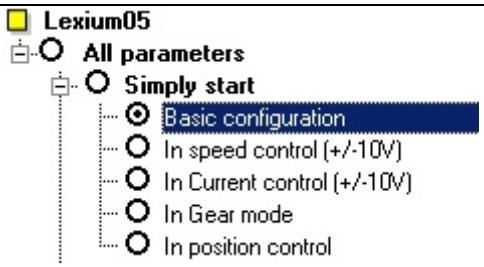
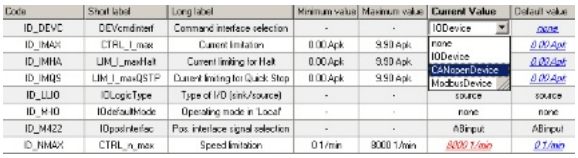
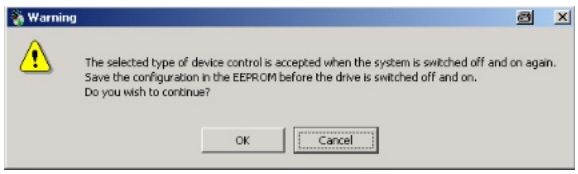
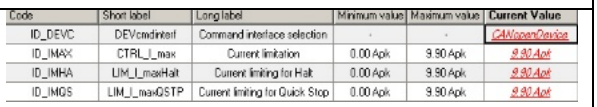
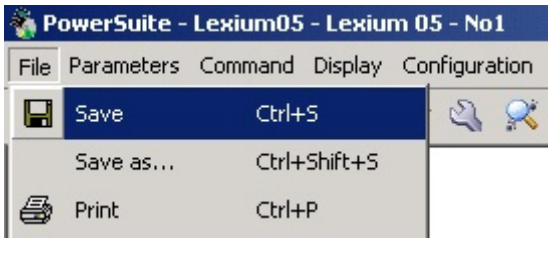
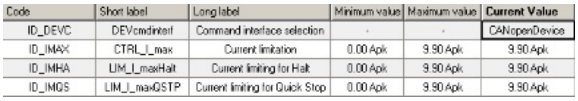
Basic settings
contd.

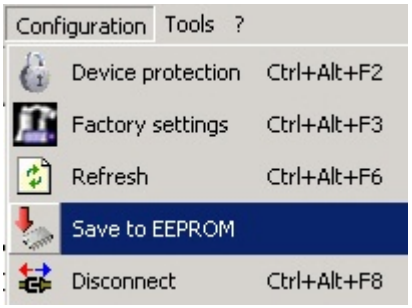
3	When the drive is supplied with 24V for the first time, or if the factory settings have previously been loaded with the PARfactorySet parameter, all the drive functions are still blocked. **You must carry out an initial setup procedure.** To connect the device to the CANopen master you must set up: - Control mode - Signal Selection Position-interface - CANopen parameters - I/O Logic type On completion the drive should report “RDY” (ready) in the status display.	

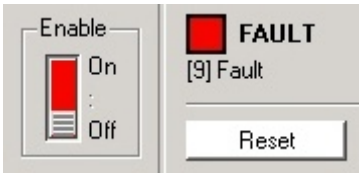
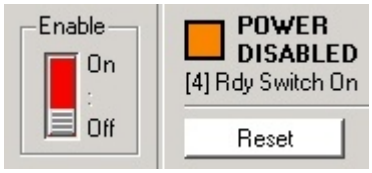
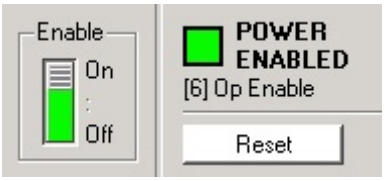
PowerSuite mit LXM05

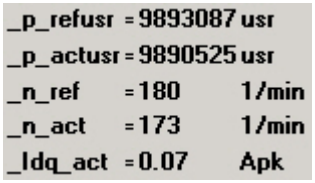
Besides using the control panel on the device, you can also use Powersuite to configure the Lexium05.		
1	Use **Action->Connect** OR The icon in the toolbar to connect to the Lexium05. Remember to connect the PC to the Lexium05 with the correct cable.	

2	Once connected, PowerSuite recognises a new device and asks for confirmation to insert it into its database. Acknowledge the request with **Create**	
3	Enter a **new name** for the device when requested and exit with **OK**.	
5	The data is read out from the Lexium05.	
6	Once the transfer is complete, the device data will be displayed.	
7	You can select the relevant drive in the project browser on the left-hand side.	

8	The parameters can be displayed in list format or in page view.	
9	Select: **Simply start->Basic configuration**. In the field for defining the control method for the device, you should select **CANopenDevice**. The servo drive will now be enabled for control via CANopen. In order for this change to take effect on the Lexium05, you will need to switch the device off and then back on again.	  
10	Initially, the change will be highlighted in red, but the display colour will change when you select: **File->Save**.	  

11	Make the following settings under Communication: **CANopen address: 3...4** **CANopen baud rate: 500** **Modbus address: 1**	 Code | Short label | Long label | Minimum | Maximum | Current Value ID_ASIH | IO_AutoEnable | Automatic Enable at PowerOn | - | - | off ID_COAD | CANadr | CANopen address (node number) | 1 | 127 | 3 ID_COBD | CANbaud | CANopen baud rate | - | - | 500KB ID_MBAD | MBadr | Modbus address | 1 | 247 | 1 ID_MBBD | MBbaud | Modbus baud rate | - | - | 19.2KB ID_MBFO | MBformat | Modbus data format | - | - | 8Bit EvenParity 1Stop ID_MBWO | MBdword_order | Modbus double word sequence | - | - | HighLow ID_SMC | DCOMcompatib | Transition 3->4 (DriveCom) | - | - | Automatic
12	To transfer the settings to the Lexium05, select: **Configuration ->** **Save to EEPROM**.	
13	The settings will now be transferred. To ensure that the settings are saved on the servo drive, you need to confirm the prompt by clicking **OK**. PowerSuite confirms the save when completed	

14	You have the option of controlling the servo drive via the PowerSuite software. To be able to do this, you must first set the **Command** switch to **Active**. Press **Alt+F** to confirm the security warning.	
15	Now set the **Enable** switch to **On**. Potential errors can be acknowledged by clicking **Reset**.	  
16	**Test run** can be used to activate the servo drive. Test stop can be used to stop it again.	
17	Use the buttons **Neg** and **Pos** to change the direction of rotation on the drive.	

18	Information about the speed and position is displayed on the bottom right.	

Performance

Scan and Cycle time

Using the described configuration and the example application code, a cycle time of 30 milliseconds was not exceeded. The memory usage in this system with the specified PLC was 35% for system data and 25% for program data.

Appendix

Detailed Component List

Hardware-Components				
Pos.	Amt.	Description	Part Number	Rev./ Vers.
1.01	1	Master Switch	VCF02GE	
1.02	1	Power supply 240VAC / 24VDC-3A	ABL7RE2403	
2.01	1	Modular Device with 40 on board I/Os	TWDLMDA40DTK	SV3.2
2.02	1	RS 485 Serial Connection Module	TWDNOZOD485D	
2.03	1	CANopen Master module	TWDNCO1M	
2.04	1	Analog Module with 2 Inputs	TWDAMI2HT	
2.05	1	Analog Module with 1 Output	TWDAMO1HT	
3.01	1	Bus coupler CANopen NIM	STBNCO2212	
3.02	1	Power supply. 24VDC PDM	STBPDT3100	
3.03	1	Relay module 2x 24VDC / 2A	STBDRC3210	
3.04	1	Digital Inputs module 6x 24VDC	STBDDI3610	
3.05	1	Analog Outputs module 2x 0...20mA	STBACO1210	
3.06	1	Analog Inputs Module 2x 0...20mA	STBACI1230	
3.07	0	Bus Terminator (incl. with Bus coupler)	STBXMP1100	
3.08	1	Connector PDM 18MM	STBXBA2200	
3.09	1	Connector I/O TYP1 13.5MM	STBXBA1000	
3.10	3	Connector I/O TYP2 18MM	STBXBA2000	
3.11	1	Connector NIM 2 Screw Type (10 St)	STBXTS1120	
3.12	1	Connector PDM 2 Screw Type (10 St)	STBXTS1130	
3.13	1	Connector I/O 5 Screw Type (20 St)	STBXTS1110	
3.14	1	Connector I/O 6 Screw Type (20 St)	STBXTS1100	
4.01	1	Magelis HMI; TFT LCD 256 colours, 5,7 inch	XBTG2330	
4.02	1	Communications Cable	XBTZ968 1.2	
4.03	1	cable adapter	XBTZG999	
5.01	2	Lexium05 Servo Drive	LXM05AD10M2	V 1.1
5.02	2	Servo motor	BSH0702P02A2A	
5.03	2	Motor Cable - 3m	VW3M5101R30	
5.04	2	Encoder Cable - 3m	VW3M8101R30	
6.01	2	Altivar 31 drive	ATV31H018M2	V1.7
6.02	1	Connection Cable for drive	VW3A8104	

Software-Components				
Pos.	Amt.	Description	Part Number	Rev./ Vers.
1.01	1	Software TwidoSoft incl. Cable	TWDSPU1001V10M	V3.2
1.02	1	Software ADVANTYS incl. Cable RS232 (STBXCA4002)	STBSPU1000	V2.0
1.03	1	Software Vijeo Designer incl. serial Cable XBTZG915	VJDSSDTGSV43M	V4.3
1.04	1	PowerSuite	VW3A8104	V2.3

Component Protection Classes

Positioning	Component	In Field, On Site			Cabinet		
					Front		Inside
Protection Class		**IP54**	**IP65**	**IP67**	**IP55**	**IP65**	**IP20**
	Mains Switch, with or without undervoltage protection and integrated indicator						**X**
	Emergency off switch housing XALK		**X**				
	Motor Protection Switch, all types						**X**
	contactors LC1						**X**
	XALD push button housing		**X**				
	Phaseo Power Supply 24 V DC/1,2 A						**X**
	Lexium 05 servo drives, all ratings						**X**
	Altivar 31 Variable speed drives, all ratings						**X**
	Twido PLC						**X**
	Advantys STB distributed I/O island						**X**
	Magelis XBTG Graphics displays					**X**	
	Servo Motor		**X**				
	CANopen TAPS with CANopen cable						**X**

Component Features

Components

Twido PLC TWD LMDA 40DTK

The modular series consists of five power bases having different processing capacities and different numbers and types of inputs and outputs (20 or 40 inputs/outputs with screw-type terminal connections or HE10 connectors, with sink/source transistor or relay outputs). The power bases can be fitted with all I/O modules (18 digital and analog modules). The supply voltage for all Twido Modular models is 24 V.

Local digital I/O:	24I/16O
Local analog I/O:	1I, 0-10 V 8 bit (512 points) 1 potentiometer on front panel. Range 0-1023 points
Application memory:	3000 instructions 6000 with memory card
Integrated interface:	Modbus RS485
Programming:	TwidoSoft

Magelis XBTG2330 Graphic Touch Panel

Display type:	LCD TFT 256 colors
Display size:	5.7" (320x240)
Protocols:	Unitelway , Modbus, Modbus TCP/IP
Interfaces:	RS232C/RS485, Ethernet 10BaseT
Voltage:	24 V = external

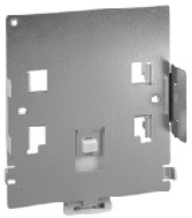
Drive Control LXM05AD10M2

Power output:	From 0.75 kW (Construction size 1)
Voltage types:	230 V ~, single-phase
Fieldbus interface:	CANopen
Signal interface:	with two analog +/- 10 V inputs and 8 digital inputs/outputs
RS 422 interface:	for pulse/direction or A/B signal inputs or encoder simulation
Operating mode:	Current control, speed control, electronic gears, point-to-point operation, speed profile, referencing, manual running

Components Contd.

Altivar variable speed drive: ATV31H018M2

- 0.37 kW, 380 to 500 V AC three-phase
- Integrated class B EMC filter
- Temperature range: -10 to +50°C
- Speed range from 1 to 20 (0 to 200 Hz)
- Speed control with flow vector check
- Operation via Modbus or CANopen possible
- 2 analog inputs plus 1 analog output
- Digital inputs
- 2 or 3 digital status outputs possible
- Protection of drive and motor
- Compact design, side-by-side installation also possible on a DIN rail using bracket VW3A11852



Phaseo power supply unit: ABL7RE2403 & ABL7RE2405

- 100 to 240 V AC/24 V DC
- 3A or 5A secondary, other ratings also possible
- Slimline design
- Parallel connection possible
- Short-circuit-proof and overload protected

Contact

Author	Phone	E-mail
Schneider Electric GmbH Machines and Process Architectures	+49 6182 81 2555	cm.systems@de.schneider-electric.com

Schneider Electric GmbH
Steinheimer Strasse 117
D -63500 Seligenstadt
Germany

As standards, specifications and designs change from time to time, please ask for confirmation of the information given in this publication.